Beginner’s Security Guide

Lockdown v0

Authors: UBNETDEF	Version: 1.0
Event: Lockdown v0	Released: April 1, 2016

Beginner’s Security Guide

Table of Contents

General Policies

Order of Importance

When competing, *generally* you should abide with the following order of importance:

Place	Item	Time Allotment
1	Injects	40% of your time
2	Server Security	50% of your time
3	Incident Reports	10% of your time

The reasoning behind order of importance is that you will gain more points when you complete an inject than you could lose for a few minutes of downtime. Having perfect uptime is **not** a guarantee you will win, one must strike a balance with uptime and inject completion.

Password Policies

Typically for competitions you should be following one of two paths:

1. Password Sheets
2. Password Formula
    a. Example: “ubnetdef14260” followed by the OS/Service Name

Using a password formula is usually less secure, however. If the Red Team manages to install a keylogger on your machine, or gains access to your plaintext passwords, it would be trivial to figure out the formula. Generally it is recommended to have password sheets as the solution for your password problem. One advantage to having password sheets is that you can have regular password changes (ex: every 1 hour), and every team member will know that password. One big disadvantage, however, would be that you have to keep the password sheet papers **extremely** secure. If any Red Team member would gain access to them (ex: via a picture), they will know the password to *all* your services, until you “rotate” or change your password sheet.

Professionalism

During a competition, you are emulating an IT Department, in various security scenarios. Therefore, like in the real world, you should observe the following:

- Keep your cursing to a minimum
    - Would you be cursing in front of your boss, or your boss’ boss?

- Be courteous
    - Keep your manners - say please & thank you
    - Do **NOT** yell at competition staff (or any member on any team)
    - Do not make fun of others - remember we are IT Professionals.
- Keep the correct attitude
    - Remember, you are there to test your skills and learn. While competition is good, do not go overboard. Keep a positive attitude, even if your team is not doing well.

Team Roles

Roles are a very important part of team dynamics and make a significant impact in your team’s overall success. It is important that every team member is assigned to their best possible role depending on their strengths, understands their role, and the tasks they must perform. The main roles a team should have are as follows:

- **Team Captain -** The leader of the team. Makes important decisions and makes sure each team member is doing what they should be doing. This member should/can have basic knowledge about all other roles, so they can properly manage the other members and participate in securing/hardening various machines.
- **Co-Captain/Inject Manager -** The second in the chain of command. Takes the role of leader, if the captain is otherwise pre-occupied. This role also doubles as inject manager. Inject manager is one of the most important roles. It is impossible for one person to do all injects, therefore they must be assigned by the person to the individual(s) most suited for the task. They also must remind the person of the due time and prepare injects for submission to white team.
- **Windows Admin -** This team member should have the most Windows knowledge among your team members and is responsible for overlooking Windows machines and services even if there are other team members on Windows machines. The “go-to” person for Windows questions.
- **Linux Admin -** This team member should have the most Linux knowledge among your team members and is responsible for overlooking Linux machines and services even if there are other team members on Linux machines. The “go-to” person for Linux questions.
- **Firewall/Network Admin -** The person responsible for any networking/firewall configuration and hardening for your team. This is the person on your team who has the most networking experience, and is the most important member of your team. This team member’s issues must be addressed with the highest regard because if your team loses network, all your services will go down. Their job is to manage the team’s firewall, block malicious IP’s, look for suspicious traffic and protect the internal team network.

These are the condensed roles when participating in a competition, but as teams and competitions get more advanced, so will your roles.

Interacting with Management

Some injects or tasks involve teams meeting with or interacting with a member of your company's management such as a CISO or CEO. They will want you to provide information about what you're doing, why you're doing it and why any breaches have occurred. They may ask you a variety of questions, so the team member you should send should be someone, preferably your inject manager or team captain. It would be wise to know the following:

- Your complete infrastructure
- Knowledge of any breaches
- The status of each service
- What you're doing to secure each machine/service

There should always be one main team member who you will send to interact with management. The person should be confident, calm and respectful when talking to any management. They should also have a high-level understanding of everything your team is doing; and constantly ask for updates from various team members. This person will be the gateway between your team and upper level management and could be the difference between getting your team a raise or getting your team fired.

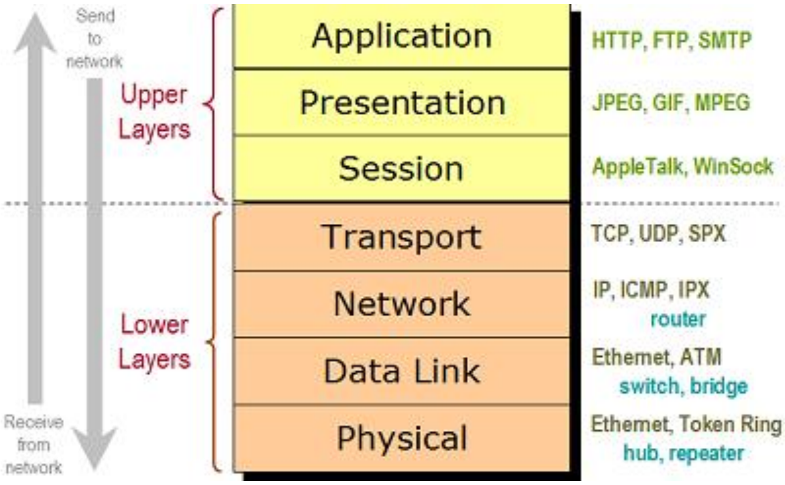
Beware of Your Surrounding Environment

In general, you should follow the following advice:

- Know your surroundings. Be aware that the Red Team/Red Team Spies might be wandering around, taking pictures of your working area or computer screen
- Look over your shoulder! Extend the same cautious to your team mates, and watch their shoulder as well.
- Keep important documents covered up, or flipped over. Only bring them to a readable state when you are using them, otherwise keep them hidden.

Troubleshooting

Generally when faced with an issue, you should follow the OSI Model and work your way up:



Verify each layer starting from the bottom (Physical), before moving up. If you still are unsure of the problem, ask one of your team members, or your team captain! S/he might have advice for you about solving your issue.

Competition Infrastructure

The competition infrastructure is different for every competition, but it is usually made up of local workstations and headless servers, which models the way machines are setup in real life.

Headless Server

A headless server is a machine without a monitor or any physical interface for you to interact with. The only way to interact with these headless machines is to log in remotely from another machine such as a local workstations. It is also possible that these machines can be located outside your local network and you have to access them remotely through the internet. This is very convenient because you can access these machines from anywhere, however it poses a security concern because it means anyone can access those machines remotely as well. **Also, there is a very real possibility that you could accidentally lock yourself out, so be careful with firewall rules and blocking SSH connections.** These servers can be providing services that your customers/business needs to run and should be treated with the utmost importance.

Local Workstation

A local workstation is a machine you have physical access to, and is mostly used to connect to headless servers and work on them. These machines are easier to secure because you can disable any remote connections and just focus on physically accessing your machine. Local workstations are usually located in your internal network, and sit behind a firewall.

pfSense Startup

You have a pfSense virtual machine in your infrastructure which is acting as your team's virtual firewall and router. This means your pfSense virtual machine is responsible for protecting your internal network, and routing machines inside your network. This machine is very important, and must be configured correctly or else your network will have a lot of problems and holes for attackers to enter through.

Introduction to pfSense

Your pfSense firewall and router settings can be accessed by two ways: your actual machine's terminal, or a web GUI that can be accessed from a workstation machine.

1. **Machine terminal:** To access your machine, just login to the pfSense VM using the default credentials that white teams have given you. This way is more complex and easier to mess up if you're a beginner and unfamiliar with the pfSense software.
2. **Web GUI (Graphical User Interface):** To access the web GUI of your router/firewall, you must open a web browser on a workstation machine on your internal network, and type in your pfSense machine's IP address into the address bar. This will bring you to the web GUI login information, which should be the default login credentials of *admin/pfsense*. **It is recommended that you mainly use the web GUI to edit your firewall settings and harden pfSense.**

pfSense Hardening

Once you have successfully logged on to your pfSense machine, you should start hardening it, like you would with every other machine on your network. Unlike every other machine in your network, what you do directly impacts the rest of the machines on your internal network. This means if you block the wrong traffic, you could potentially bring down your entire team's network. This means securing your pfSense router is a high priority because it impacts the most machines. To harden your pfSense machine, follow these steps:

1. Change your default admin password by going to **System > User Manager** and clicking on your admin account. Then change the password accordingly and apply changes.
2. Check the users and their privileges to limit which users can do what, by clicking **System > User Manager.**
3. Check the current firewall rules. To see the current rules, go to **Firewall > Rules.** You'll see different tabs of WAN, LAN, etc. Those control the different aspects of your network and should be configured accordingly. You should add firewall rules, whitelisting certain IP addresses that you know are good and ignoring everything else. However, make sure you don't accidentally block the scoring engine, as you will lost points for it.
4. Backup the current configuration. A backup can be saved by going to **Diagnostics > Backup/Restore**, and clicking **Download Configuration**. You can restore this file later if you screw up firewall rules or your configurations get messed with.

If you need additional help, or have trouble with any of these steps, you can look at the documents about pfSense located here: https://doc.pfsense.org/index.php/Main_Page

Monitoring Your Network

Now that you have your firewall setup and secure, you can start monitoring your network for any malicious traffic and block the corresponding IP addresses. To do this you will take a look at pfSense logs. These logs can be found by going to **Status > System Logs,** and then clicking which type of logs you'd like to look at (system, firewall, etc.). By viewing firewall logs, you can see which IP addresses are trying to connect to your system or any outgoing connections from your internal network and block anything that looks suspicious.

Linux Startup

At the start of the competition, you will login to your OS with the default credentials given to you by white team.

Operating Systems Overview

The first thing you need to identify when you start the competition is which Linux OS you are running, which can be done by opening the terminal and typing:

```
cat /etc/*-release
```

Once this is done, you can then lookup commands specific to that OS, such as how to change your password, put up the firewall, etc. The tables below may help you with basic commands that are different across Linux distributions.

The **package manager** is what manages downloads, updates, etc. on your machine. Depending on which Linux OS you’re running, the command changes as listed below.
Firewall commands have to do with the built-in firewall software, which need to be configured specific to whichever service you are running, and what port you are running it on.
Managing Services includes commands that control services running on your machine, including how to stop them, start them, restart them and check their status.

Ubuntu

Package Manager	APT (Command to run: `apt-get`)
Firewall	`ufw` (Interface to IPTables) `iptables`

Managing Services	Ubuntu Version >= 15.04
	`systemctl {start|stop|...} {service_name}.service`
	
	Ubuntu Version < 15.04
	`service {service_name} {start|stop|....}`

Notes	By default, the root user account is locked. You must use sudo to run any commands as root. Any user added to the group “admin” will gain access to sudo by default.

Debian

Package Manager	APT (Command to run: `apt-get`)
Firewall	`iptables`
Managing Services	`service {service_name} {start|stop|....}`

CentOS

Package Manager	yum (Command to run: `yum`)
Firewall	`iptables`

Managing Services	[CENTOS <= 6]
	`service {service_name} {start|stop|....}`
	
	[CENTOS >= 7]
	`systemctl {start|stop|...} {service_name}.service`

Notes	By default ships with a service known as “SELinux”. Similar to AppArmor on Ubuntu.

FreeBSD

Package Manager	`Pkg`
Firewall	`Ipfw`
Managing Services	`service {service_name} {start|stop|....}`

Linux Guides

Basic Linux Hardening

After you've logged in and identified your Linux distribution, the following steps should be done at the start of any competition to do, what's called, "hardening" your system. "Hardening" means to change some default settings that allow for easy access from bad guys, aka the red team. The goal when you finish running the following steps is that you will have a relatively hardened server (OS), giving you time to harden your respective service. **Some steps don't include the exact commands you must type, these are the things you will need to Google in order to learn!**

1. Change the root user's password
    a. Type: `/usr/bin/passwd`

    This is the password you use to login to the actual OS, and change any important settings on your system.
2. Enable all the firewalls!
    a. Typically, you will be unable to disable all outbound internet access. Figure out what firewall tool is installed on your operating system (See: Operating Systems, above), and enable it.
    b. Block all inbound & outbound connection by default. Depending on which firewall system you are using for your specific distribution, you will need to lookup how to do this.
    c. Enable outbound connections via port 80 (HTTP) and port 443 (HTTPS), for any updates, or software downloads
    d. Enable inbound connections to any ports required by your service(s). For more information, see Service Guides below.
    e. The final step is to reload the firewall settings so that the changes you made actually takes place.

****IMPORTANT: If your service(s) goes down after you apply firewall rules, you probably blocked something you shouldn't have. This isn't necessarily red team, this is probably you! Try to figure out what you blocked that you shouldn't have or start over.**

3. Take backups of important files
    a. Backup files that are key to the service you are running, for example, if you are running a web server (HTTP), backup your website to another file to reload it if red team destroys yours! This way you can always restore your website to its original state. Make sure you put this in a place that won't be obvious to red team, but also so you won't forget where you put it!

   b. Backup the state of the machine
      1. Take a snapshot of the processes running (via `ps aux`)
      2. Take a snapshot of the open network ports (see Checking Connections)
4. Disable unnecessary services. To do this, you must find out what services are running.
5. Disable unnecessary users. To do this, you must find out what users are on your system.

****IMPORTANT: Be careful not to delete accounts, ONLY DISABLE, as they could be a customer (scoring engine) and could be viewed as service failure if your customer can't access the system.**

6. Audit user accounts
   a. Verify the accounts on the system are required for this system. Disable any accounts
   b. Check what accounts are enabled
   c. If any user accounts have a shell (eg: bash), check if they have authorized any SSH keys (see: `~/.ssh/authorized_keys`)
7. Audit SSH configuration
   a. Sometimes, SSH (often through the software OpenSSH), is configured incorrectly. Edit the file `/etc/ssh/sshd_config` to ensure the following settings are correct
   b. Disable remote root login
      1. Set the config key "`PermitRootLogin`" to "`no`"
   c. Whitelist what users are allowed to remotely login
      1. Set the config key "`AllowUsers`" to be a list of the usernames you are whitelisting, followed by a space.

IPTables

IPTables is an application that allows a system administrator to configure rules to apply to the treatment of packets.

IPTables Command Reference

What do you want to do?	Command to run
View all defined rules	`iptables -L --line-numbers`
Clear out all defined rules	`iptables -F`
Delete a specific rule	`iptables -D {chain} {number}`
Add a rule	`iptables -A {chain} {rule} -j {ACTION}`
Apply the default policy of (ACCEPT/DROP/REJECT)	`iptables -P {chain} (ACCEPT/DROP/REJECT)`
Allow HTTP traffic in (TCP, Port	`iptables -A INPUT -p tcp --dport 80 -j ACCEPT`

80)	
Block IP 8.8.8.8	`iptables -A INPUT -s 8.8.8.8 -j REJECT`
Allow ONLY 192.168.1.10 to access MySQL (TCP, Port 3306)	`iptables -A INPUT -s 192.168.1.10 -p tcp --dport 3306 -j ACCEPT`
Allow SSH out	`iptables -A OUTPUT -p tcp --dport 22 -j ACCEPT`

Allow ICMP	`# Need both rules`
	`iptables -A INPUT -p icmp --icmp-type 0 -j ACCEPT`
	`iptables -A INPUT -p icmp --icmp-type 8 -j ACCEPT`

Possible chains are:

- INPUT: Packets destined for the host computer
- OUTPUT: Packets originating from the host computer
- FORWARD: Packets not destined or originating to/from the host. Used for routing/by routers

Possible actions are:

- ACCEPT: Allows the packet to continue
- DROP: Drop the packet, like a blackhole
- REJECT: Drops the packet, but sends a response

Possible flags for a rule:

- -s: Source IP
- --sport: Source Port
- -d: Destination IP
- --dport: Destination Port
- -p: Protocol (TCP/UDP/ICMP)

General Startup Config

Running the following commands as a superuser (root), you would achieve the following.

- Applying a default policy of rejecting packets you do not whitelist
- Allowing ping to/from our server
- Allowing SSH, HTTP, and HTTPS traffic to our server

```
# Flush any previous rules
iptables -F

# Apply a default policy of rejecting
```

```
iptables -P INPUT REJECT
iptables -P OUTPUT REJECT

# Allow servers to ping us
iptables -A INPUT -p icmp --icmp-type 0 -j ACCEPT
iptables -A INPUT -p icmp --icmp-type 8 -j ACCEPT

# Allow us to ping servers
iptables -A OUTPUT -p icmp --icmp-type 0 -j ACCEPT
iptables -A OUTPUT -p icmp --icmp-type 8 -j ACCEPT

# Allow the following ports:
# 22(SSH), 80(HTTP), 443(HTTPS)
iptables -A INPUT -p tcp --dport 22 -j ACCEPT
iptables -A INPUT -p tcp --dport 80 -j ACCEPT
iptables -A INPUT -p tcp --dport 443 -j ACCEPT
```

After Linux Hardening

Once you have finished the hardening guide, and your firewall rules are working and correct, your main priority become completing injects that your inject manager assigns to you and finding/keeping red team out! This can be a difficult task, especially since these competitions are designed to make you work hard to keep your system secure, just like in real life! Located below are some main things you should monitor/check to look for and remove any traces of red team in your machine.

1. **Check connections to your machine:** To list active connections to your machine (especially the ones you don’t want), we would be using the handy dandy `netstat` tool. Usage:

```
netstat -tupn
```

Explanation of the flags:

- -t : TCP Connections
- -u: UDP Connections
- -n: Do not resolve IP Address to DNS names
- -p: What program is responsible for this connection

Alternative command: `lsof`. Usage:

```
lsof -nPi
```

Explanation of the flags:

- -n: Don’t convert IPs to DNS names
- -P: Don’t convert port numbers to port names
- -i: Interface. We’ll leave this blank (we want TCP & UDP). We could put either “TCP” or “UDP” here.

2. **Check open ports:** Ports are the windows to your computer, if left open, anyone can enter them or leave malicious materials running on them. To see what ports are have open, we again would be using the `netstat` tool. Usage:

```
netstat -tunlp
```

For an explanation of the flags, see above. We add the following flag:

- -l: Show listening ports

3. **Check permissions and attributes:** Typically, to find permissions for a directory or file we would use the `ls` command. Combined with the “-l” flag, we would get some sort of output similar to below

```
root@server:~# ls -l
total 4
-rw-r--r-- 1 owner-username group 743 Feb 17 22:17 random.file
```

Refer to the Permissions Overview section for an idea what each of these letters means. Each color-coordinated group of characters is the permissions for either the owner/group/everyone else.

- Red (First 3 chars, from right): Permissions for everyone else
- Green (Next three characters): Permissions for the group that owns this file/folder
- Grey (Last three characters): Permissions for the owner of the file

To look at any special attributes on any file, use the `lsattr` command. Usage:

```
root@server:~# lsattr temptest
-------------e-- temptest
```

(e) is normal. It means extent format. To remove an attribute, use `chattr -{ATTR} {FILE}`. Likewise, to add an attribute use `chattr +{ATTR} {FILE}`.

4. **Check users:** All users that have access to this system will be in the file called “`/etc/passwd`”. Some may not be malicious, but this doesn’t mean they need the same permissions as our super user. We need to make sure every user only has the permissions they need. For more information on reading this file, see the table below.

5. **Check important file locations:** The following will be a quick reference of important files and locations that you should check for anything that looks out of the ordinary. If you’re not sure what “ordinary” looks like, look it up!

****NOTE****: When attempting to view untrusted files, do **NOT** use `cat`. Use the text editor `nano`

Location	Notes
~/.ssh/authorized_keys	This file contains what public SSH keys are allowed to login to this user. If there is an SSH key match on login, the user is not required to enter a password.

/etc/passwd	This is the password file. It will show you all valid users on the system. The most important bit is the end of each line, it will sometimes say:
	`/bin/bash`
	Or some other shell. If if has something there, it is usually a good indicator that that user can login. One way to disable a user's account would be setting the user's shell to:
	`/bin/false`
	(To do that, run `chsh -s /bin/false USERNAME`)

/etc/shadow	This file contains the passwords of all users on this system, assuming they have one. Permissions wise, **ONLY ROOT SHOULD BE ABLE TO READ/WRITE TO IT**. The following is generally what the permissions/file ownership should look like:
	`-rw-r----- 1 root shadow 1038 Mar 4 21:15 shadow`
	Ensure that on all the systems you manage the shadow file looks like above.
	To verify a user account is disabled, simply view the shadow file as root.
	`username:!:16063:0:99999:7:::`
	The part highlighted in red denotes the encrypted user's password. Generally if this is a "!", "*", or any single character, it means that the user's account is disabled and cannot login.

/etc/sudoers	If the `sudo` command is installed on your server, this file manages with users are allowed execute commands as a superuser (root). To edit this file, use the command `visudo`. Ensure that you know exactly what users are listed here. They will have the ability to execute any command as a

	superuser (root).
/etc/rc.local	This script is executed at boot. Sometimes you may find “nasty” stuff in here. Be sure to at least look it in the beginning of the competition.
/etc/inittab	Similar to the file “rc.local”, however this script executes commands based on runlevel. Look for anything that says “respawn” in it’s line. When that process is killed, it will automatically be restarted.
/var/log/auth.log	This file lists all successful and unsuccessful authentication attempts. Monitor this file for anything nasty happening. Examples would be: ● Multiple failed login attempts for a single IP ● Login attempt for a system-level user (ex: cron), or any user not known/whitelisted ● Any not known logins to root that **YOU** did not do.
/var/log/syslog	Any log sent to the system, except for authentication logs, will be located here.
/var/log/messages	Generic system log. This log contains the “general system activity”, that are considered non-critical.
/var/log/kern.log	Any messages from the system kernel will be sent here.
/etc/crontab	This is the system-wide crontab file. Commands in this file will be executed at a regular interval (defined in the file). Monitor this file, along with other user’s cron. Usually this is a good place to hide any remote shells, or worse.
/etc/cron.d /etc/cron.daily /etc/cron.hourly /etc/cron.weekly /etc/cron.monthly	These are the rest of the directories that store cron information. Be sure to check all these directories for any possible malicious entries.

/var/spool/cron/crontabs	User’s individual crontabs are listed here. Another way of viewing user’s crontabs would be to call the following command:
	`crontab -u {USER} -l`
	To edit a user’s crontab, run:
	`crontab -u {USER} -e`

/tmp	This is linux’s default temporary file location. Typically, there should be no executable files located in this directory.

6. **lsof** - sysadmin’s Swiss Army knife: Most information for this section was taken from an article by Daniel Miessler.[1] Using this program will help you identify and resolve any red team

Showing all connections

```
# Showing all
root@server:~# lsof -i

# All TCP
root@server:~# lsof -iTCP

# All TCP connections to a certain port (ex: 22)
root@server:~# lsof -iTCP :22
```

Show which files a user has open

```
# Show all files a user has open
root@server:~# lsof -u USERNAME

# Kill all processes owned by a user
root@server:~# kill -9 `lsof -t -u USERNAME`
```

Show open files with a link count of less than 1

This command will have an output if an attacker is trying to hide something

```
root@server:~# lsof +L1
```

List all open files

```
# Everything
root@server:~# lsof

# Everything, except owned by the user root
root@server:~# lsof -u ^root
```

[1] https://danielmiessler.com/study/lsof/

Windows Guides

Basic Windows Hardening

You might have more experience with Windows, especially since it's a lot more user friendly than Linux distributions. However, you still need to called, "harden" your system as much, if not more than your Linux machines. "Hardening" means to change some default settings that allow for easy access from bad guys, aka the red team. The goal when you finish running the following steps is that you will have a relatively hardened operating system, giving you time to harden your respective service.

1. Change the [Domain/Local] Administrator password:
    a. Press Ctrl+Alt+Delete and choose Change Password
2. Enable Windows Firewall
    a. Control Panel>Windows Firewall. Make sure it is enabled and set to a "public" profile. Check the firewall rules for malicious configurations, and only allow ports that contain scored services. Then re-enable the NIC.
3. Open Task Manager. Make note of any malicious processes and terminate them. **Typically Viruses will have a blank description in Task Manager.**
4. Harden Active Directory
    a. Open AD. Hold down the windows symbol and "R" then type "dsa.msc", press OK. Scan AD for any suspicious accounts, and disable. Check the membership of Administrators, Domain Admins, Enterprise Admins, and so on. Disable any suspicious accounts.
    b. Raise the Domain Functional Level to the highest available (right click on Domain and choose properties)
5. Harden services (Run "services.msc"). Disable or stop any suspicious or unused services. You might need to research some.
6. Check for Shared Files or Folders.(Run "compmgmt.msc"). Ensure all file shares are removed. While in Computer Management, check Local Users and Groups for any suspicious accounts and disable them
7. Disable RDP and Remote Assistance (Control Panel>System).
8. Remove any Scheduled Tasks or Events. Search for Task Scheduler.
9. Install Peerblock, and begin monitoring incoming connections, blocking where appropriate.

So You Want to Use Windows Firewall

It's easy! Windows Firewall provides a simple, GUI-oriented way to setup a host based firewall for your local machine. This adds an additional layer of protection on top of whatever is running at the network hardware level (i.e. pfSense)

1. To Access Windows Firewall, go to the Start Menu > Control Panel > Windows Firewall.
2. Make note that Windows Firewall segments its rules based on Home/Work Networks and Public Networks. Each table will give a quick summary of how it treats inbound connections, and if there are any active networks that fit that category.

   a. The default inbound connection rule should read "Block all connections to programs that are not on the list of allowed programs". There is much more to configure but this basic rule will help aid against a plethora of attacks and hail mary's by the red team.

3. On the left side of the Windows Firewall screen is an option to turn Windows Firewall on or off - if the firewall is not already on make sure to click this!
4. For simple ruleset editing, click "Allow a program of feature through Windows Firewall". Scan this list for any programs that should probably not be there - i.e. servers that have no business running on that particular machine, programs with weird names/descriptions, etc.
5. For more advanced editing, and yes you'll have to do this in a competition scenario, go to "Advanced Settings" on the Windows Firewall screen.
   a. From here, you can edit specific inbound and outbound connection rules. Pay close attention to these, and Google search any whose meanings aren't obvious. Some that you may want to disable might include Core Networking - IPv6 under inbound connections ONLY if you are sure that you don't need IPv6 in the competition. This is help to shut off when not needed since it provides a fairly large attack vector for the red team.

Windows Log Files

Using Windows Logging will give you a lot of insight as to what is happening on your machine. To access logs from Windows, do the following:

1. Go to Start > Control Panel > Administrative Tools > Event Viewer
2. On the left side of the Event Viewer, expand the "Windows Logs" tree.
   a. Here you will see five categories - Application, Security, Setup, System, and Forwarded Events.
   b. These are all fairly self-explanatory, however Forwarded Events is worth discussing in more detail. Windows allows machines to send and receive the log files from the other four categories in order to set up centralized logging. This is a great feature to use in a competition where setting up a solution such as Splunk may not be a wise use of limited time. Also, this specific feature may show up in injects in competitions, as it did in the NECCDC. In order to set this up, there are a few things that you'll need to do:
      i. <Insert event forwarding instructions here>
3. Click on "Security" to view security-related logs.
   a. Many of these logs focus on logon attempts which can be useful if you think that the red team has access to your machine and is logging on and off of it under your nose. Keep an eye on the "Audit Success" and "Audit Failure" keywords to see if login attempts have succeeded or failed, as well as their time and any comments put in the log entry by Windows.

b. You can also export log entries into an XML File - this can be useful for parsing by various logging solutions and also can be great for incident report creation/white team briefings.
c. The other log categories function similarly, read through them and see what you find!

Useful Windows Tools

1. Process Hacker - http://processhacker.sourceforge.net/downloads.php
2. Sysinternals (specifically TCPView) - https://technet.microsoft.com/en-us/sysinternals/bb842062
3. Microsoft Enhanced Mitigation Experience Toolkit - https://www.microsoft.com/en-us/download/details.aspx?id=46366
4. Microsoft Baseline Security Analyzer - http://www.microsoft.com/en-us/download/details.aspx?id=7558
5. psexec
6. sysmon - http://www.darkoperator.com/blog/2014/8/8/sysinternals-sysmon

Services Guides

As a blue team member, you will be responsible for maintaining a service(s) for your team, and keep it running for your business to make money (gain points). You must identify which team member is responsible for which service and work with the corresponding machine already running that service. There are many different types of services your machine can run, but the main ones that are part of most competitions are: **HTTP (web), DB (database), FTP (File Transfer Protocol), AD (Active Directory), DNS (Dynamic Name System), SSH (Secure Shell), RDP (Remote Desktop Protocol), ICMP (Internet Control Message Protocol),** etc. You must interact with these services to make sure they're running, working and secured properly so red team can't break them.

Apache - HTTP/HTTPS (Your Website)

Apache is a web server software, used to delivery pages via HTTP/HTTPS. This is usually the website your business is running for your customers to access materials or buy products. If it's down, your business loses credibility, customers, income, etc.

Required Ports
80/tcp, 443/tcp

Config File Location
Please check one of the following directories:

- /etc/apache
- /etc/apache2
- /etc/httpd

Hardening Guide
- Ensure Apache is running as a non-privileged user (typically it's own user account)

- Ensure the web directory (typically /var/www/html) has "sensible" permissions. This means:
  - Directory should be owned by the webserver's user (typically apache)
  - Directory should only be writable by the owner

Active Directory

Active directory is a "central" database containing people's information that will be used to authenticate them. Every Active Directory structure has to one a domain controller. A domain controller a server that takes care of managing Active Directory, including hosting its database and handling the authorization, authentication and accounting mechanisms. Domain controllers typically run Windows Servers. You can successfully control users and computers that are joined to your domain and apply group policies to clients that are a part of your domain, including password policies.

FTP (File Transfer Protocol)

An FTP server is used to transfer files between machines, and can contain sensitive company documents, so it is imperative that only the proper people should be able to view, access and modify these documents.

Required Ports
21/tcp

Configuration File Location
/etc/vsftpd.conf

Hardening Guide
- Disable anonymous access
- Enable logging to keep track of user activity

RDP (Remote Desktop Protocol)

When RDP is enabled on your machine, it allows a user to remotely login and use the machine like a local workstation. It shows the full desktop and allows the user to interact with the machines like they were sitting right in front of it. While this is convenient for remote administration purposes, it must be secured correctly to prevent malicious users from using it.

Required Ports
3389/tcp

Hardening Guide
- Check which users can RDP
- Limit users who can RDP to your machine
- Specify permissions using group policies
- Set an account lockout policy

MySQL (Database)

MySQL is a database server often used in conjunction with web applications for storing persistent data.

Required Ports

3306/tcp - ONLY IF ENABLING REMOTE MYSQL

Configuration File Location

/etc/my.cnf

Hardening Guide

- First run the command: `mysql-secure-installation`
- When granting remote mysql, be sure to only enable the user for a specific IP address, and never grant DROP privileges to any user but root.

Other Notes

All these commands will require you to enter the mysql command line interface. To enter the command line interface, type `mysql -u root -p`

How to add a user:

1. Enter the following command at the mysql prompt

```
CREATE USER 'USERNAME'@'IP.ADDRESS.HERE' IDENTIFIED BY 'SECRET-
PASSWORD';

GRANT SELECT,INSERT,UPDATE,DELETE ON 'database'.* TO
'USERNAME'@'IP.ADDRESS.HERE';

FLUSH PRIVILEGES;
```

Viewing all users

1. Enter the following command at the mysql prompt
   ```
   SELECT User, Host from mysql.user;
   ```

Viewing all privileges

1. Enter the following command at the mysql prompt
   ```
   SHOW GRANTS FOR 'USERNAME'@'IP.ADDRESS.HERE';
   ```

SSH Server - OpenSSH

OpenSSH is an SSH server which allows you to remotely connect to another server you have login access to

Required Ports

22/tcp

Installation Guide

Most systems have OpenSSH installed by default. If not, installing OpenSSH is as simple as using your operating system’s package manager and installing the package “openssh”

Config File Location

To configure the OpenSSH server, edit the file:

- /etc/ssh/sshd_config

Hardening Guide

- Ensure “`PermitRootLogin`” is set to “`no`”
- Add “`AllowUsers`” to the bottom of the configuration file. Add the usernames you wish to whitelist, separated by a space. Example:
  `AllowUsers user1 user2 user3`
- Create a file called `sshrc` inside the `/etc/ssh` directory. Put the following contents:

```
# Get the users IP
ip=`echo $SSH_CONNECTION | cut -d " " -f 1`

# Log the login
logger -t ssh-wrapper $USER login from $ip

# Add the login to a file
# Change /tmp/ssh_logins.txt to a file you will monitor
echo "Login for $USER from $IP" >> /tmp/ssh_logins.txt
```

Cheat Sheets

Common Ports

Service Name	Port
21	FTP
22	SSH
25	SMTP
53	DNS
80	Web
88	Kerberos
110	POP3
123	Windows Time
137	NetLogon, NetBIOS Name Resolution
139	DFSN, NetBIOS Session Service, NetLogon

143	IMAP
389	LDAP
443	Web
445	SMB,CIFS,SMB2, DFSN, LSARPC, NbtSS, NetLogonR, SamR, SrvSvc
464	Kerberos Password Change
636	LDAP SSL
993	IMAP SSL
995	POP3 SSL
3268	LDAP GC
3269	LDAP GC SSL
3306	MySQL
3389	Remote Desktop Connection