Internet Radio Linking Project (IRLP) Wiki

Keeping the Radio in Amateur Radio

Securing Remote Secure Shell (SSH) Access

If you need to login to your node using SSH remotely, the following will help you avoid unauthorized access to your node.

Exposing the default SSH port 22 to the internet and leaving root logins enabled (the default in the CentOS IRLP build) is not a good practice and will attract a constant barrage of attempts to gain access to your system. If you have a weak root password set, it can be a matter of hours until something or someone has gained access.

Strong Passwords

Using strong passwords is a good idea, if you have trouble remembering strong passwords make a note of them in a notebook or other non-electronic form, or in such a way as not to identify what it's for.

Online Password Generator [http://onlinepasswordgenerator.com/]

Disable Root Logins

By default with SSH you can login directly using the root username and password. Since it is a given that every Linux and Unix system has a user called root, this is the obvious choice for an attacker to use and it gives them super user access with no further effort.

We can disable logging in directly as root via SSH, after this change you'll need to login as a standard user and if root privileges are needed issue `su -`, you'll then be prompted for the root password. Once you have finished `Ctrl D` to exit back to the previous user. From the console - that is the keyboard and monitor connected to your node, you can still login directly as root.

1. As root edit `/etc/ssh/sshd_config`, find and change the `PermitRootLogin` option to read `PermitRootLogin no`. Save, and restart sshd with `/etc/init.d/sshd restart`.
2. Now you can either set a password on the IRLP `repeater` user or add your own user account. To (re)set the repeater password as root issue the following command `passwd repeater` and follow the prompts. You will now use this username and password when using SSH to login.
3. If you are doing this change remotely, open another SSH session and test logging in as repeater or with your own user account, and switching to the root account before you close your current session. Otherwise you could potentially lock your self out of the system until you can gain access to the console to straighten things out.

It is also a good practice to avoid using the root account unless you really need to be the super user to do something, one typo can hose an entire system before you know it's even happened. I managed to wipe most of the file system on a Unix system once, luckily it was on a test system, and with the OS install CD and yesterdays backup tape in hand I had the system restored to its previous state with-in a couple hours. If you have no disaster recovery plan, you can spend hours or even days piecing a system back-together.

Port 22

The default SSH port 22 is where anything and everything will try when looking for Linux and Unix hosts to compromise.

If you currently have port 22 open to the world, as root `tail -f /var/log/secure` to see who passing by has been “knocking on your door”, you'll more than likely see attempts scrolling by. `Ctrl C` to exit.

Using a non-standard port will avoid most of the attention. In this example we'll use port 22500. Any high port number of your choice is generally OK.

There are two ways of doing this:

1. In some routers, the port forwarding configuration allows you to redirect target port. For example port 22500 externally can be mapped to port 22 internally to your IRLP system.
2. If your router does not support redirecting the target port, the configuration of sshd can be changed to listen on a non-standard port. As root edit `/etc/ssh/sshd_config`, find the line with `Port 22` and under it on a new line add `Port 22500`. Save the file, and restart sshd `/etc/init.d/sshd restart`. Restarting sshd will not disconnect an existing session.
3. In each case preserve the port 22 forwarding you may already have in place, and add an additional port forward for the new port and test it by updating the settings in your SSH client and open a new session. Once you're happy everything is working as expected you can remove or disable forwarding for port 22, then check you can no longer connect on port 22 to be sure.

securing_remote_ssh_access.txt · Last modified: 2013/01/28 09:55 by 142.103.194.1