SEC 150 Assignment: Dropbear SSH Server

Titus Barik (tbarik@staff.waynecc.edu)

Version 1.0

1 Assignment Description

In this assignment, you will implement a functional SSH server using the embedded SSH server and client Dropbear[1], available as a Tiny Core package. To do so, you will generate a SSH-2 RSA public/private key pair. Roughly speaking, the public key is sharable but the private key must never be provided to untrusted entities.

Upon completion of this assignment, you will be able to login to remote systems using secure SSH keys, rather than passwords. The ability to have password-less (but still authenticated) logins are important for tasks such as nightly automated backups, where a physical user will not be present to enter a password.

More importantly, in many systems the firewall will disable all incoming ports except for SSH. In this case, the user must first authenticate with SSH and then use a technique called SSH tunneling to access all internal systems.

2 Pre-Requisites

Before starting this assignment, you should have Tiny Core up and running. If you've done the midterm project, you can continue to use that VMWare image. You will also need to install the `dropbear` Tiny Core package.

3 Instructions

1. After installing the necessary package, start the `dropbear` server using the command `/etc/init.d/dropbear start`.

2. Set a password of `sec150` on the `tc` account, using `sudo passwd tc`. This will allow you to make configuration changes to SSH before SSH keys have been deployed.

3. Using PuTTY, verify that you can SSH to the server using a username and password. Recall that `ifconfig` will provide you with the IP address of the Tiny Core machine.

[1] `https://matt.ucc.asn.au/dropbear/dropbear.html`

4. Use the PuTTY Key Generator to generate a SSH-2 RSA key pair. Save the private key (which includes the public key automatically) as `lastname.ppk`.

5. On the remote server, create a directory called `.ssh` under `/home/tc`. In this directory, create a file called `authorized_keys`.

6. In `authorized_keys`, paste the public key that you generated using PuTTY Key Generator. **Hint:** Make sure that the resulting paste is a single line. Most errors result from having new line characters in the authorized keys output.

7. Next, you'll need to fix the permissions of the files and directories that you've created:

   ```
   chmod 700 /home/tc/.ssh
   chmod 600 /home/tc/.ssh/authorized_keys
   ```

   Doing this step incorrectly will usually result in "Server refused our key" messages.

8. Now, you'll need to create a PuTTY session and configure it. Most of what you need can be found under Connection > SSH > Auth. Here's you'll want to point "Private key file for authentication" to the `lastname.ppk` that you created in this assignment. You might also want to go under Connection > Data, and set the auto-login username to `tc`.

4 Milestone

Submit a screen capture showing both Tiny Core and PuTTY on your system. The Tiny Core window should show that the `dropbear` package has been installed.

5 Submission

Here, you'll need to create an archive (`tar` stands for tape archive) of the `tc` home directory, using:

```
tar -cvf /tmp/lastname_tc.tar /home/tc
```

This will bundle up **all** the files in your home directory (including the `.ssh`) files. Submit this archive, which will be located in `/tmp`.

Also, submit your `lastname.ppk` file.

In total, you'll have two files for your final submission.