Cryptography Exercises: ccTLD Workshop: Amman, Jordan

November 27, 2007

Exercises

Using SSH public/private Keys for Authentication

Notes (CRITICAL)

1. The "#" and "$" characters before commands represents your system prompt and is not part of the command itself. "#" indicates a command issued as root while "$" indicates a command issued as a normal user.
2. ***italics***: Items that are in *italics* are to be replaced with something of your choice. For instance, *username* means choose your own username, don't literally choose the word "username".

1.) Munging a Document and Comparing Message Digests [Top]

To do this exercise you will need to be root.

On your machine type:

```
# cat /etc/motd
```

Look at your neighbour's machine. Is their file exactly the same as yours? Can you be sure?

Now run the file through the md5 one-way hashing function:

```
# md5 /etc/motd
```

Now change ONE (1) character in your /etc/motd file and repeat the md5 test. You may want to do this using two terminals. One to have your md5 output displayed and the other for editing the /etc/motd file.

Example:

```
# vi /etc/motd
```

One character change.

Compare the results with your neighbor, or with your previous md5 message digest. They should be very different.

You can repeat this exercise using `sha1` if you wish.

2.) Generate your Public/Private Key Pair [Top]

Note: *Please* be sure that you are logged in and using your admin account for this exercise - not root.

We will now generate a single DSA SSH protocol 2 key of 2048 bits. To do this, issue the following commands.

```
$ cd
$ ssh-keygen -t rsa -b 2048
```

You will be prompted for a file location for the key as well as for a passphrase to encrypt the key file. This should look like:

```
Generating public/private rsa key pair.
Enter file in whic to save the (/home/admin/.ssh/id_rsa):   [PRESS ENTER]
Created directory '/home/admin/.ssh'.
Enter passphrase (empty for no passphrase):     [TYPE IN PASSPHRASE]
Enter the same passphrase again:                [TYPE IN SAME PASSPHRASE]
...
```

Be sure to enter a passphrase. Private key files without passphrases are a security hole. Your passphrase can be pretty much anything you want and as long as you want - including spaces.

Make sure that you pick the default location to store your

You will see information about where your private and public key have been saved. Your private key should now be protected by a passphrase. This means to use your public/private key combination you will need to type in your passphrase (not your admin account's password) when prompted.

3.) Copy Your Public Key to Your Neighbor's admin Account [Top]

First connect to your neighbor's machine as the userid *admin* using ssh. We'll refer to your neighbor's machine as *pc1*.

NOTE: Do Not connect to pc1 unless you are on pc2. You should connect to the machine your instructor indicates (ask him if he has not done this!).

Here's what you do (as a normal user):

```
$ ssh admin@pc1
```

Now you'll be faced with a prompt similar to this:

```
The authenticity of host 'pc1.cctld.gy (192.188.252.201)' can't be establishe
RSA2 key fingerprint is 60:f7:04:8b:f7:61:c4:41:6e:9a:6f:53:7d:95:cb:29.
Are you sure you want to continue connecting (yes/no)?
```

You should say `yes` to this prompt, but you should understand what this means. Do you? If not, please ask your instructor, and for this class he'll tell you that we'll learn about this step in more detail tomorrow :-)

Once you say `yes`, then you see another message like this:

```
Warning: Permanently added 'pc1.cctld.eu.org' (RSA2) to the list of known hos
[/etc/ssh/ssh_host_key.pub]
admin@pc1.cctld.eu.org's password:
```

At this point enter in the password for the admin account on your neighbor's machine.

Now you'll be logged in and see a prompt like this:

```
[admin@pcN ~]$
```

Now you should logout of your neighbor's machine, and then immediately log back in:

```
[admin@pcN ~]$ exit
$ ssh admin@pc1
```

Now you should simply be prompted for the admin password on your neighbor's machine. You

should not see the warning message again. Now, log out of your neighbor's machine again:

```
[admin@pcN ~]$ exit
```

Let's copy the public_key for your user account on your machine to the /usr/home/admin/.ssh directory on your neighbor's machine. As usual there are several ways to do this, but here's one set of steps that should work (be sure you are on *your* machine):

```
$ cd /usr/home/admin/.ssh
$ scp id_rsa.pub admin@pc1:/tmp/.
$ ssh admin@pc1
[admin@pc1 ~]$ cd .ssh [if ".ssh" is not there do "mkdir .ssh"]
[admin@pc1 ~]$ cat /tmp/id_rsa.pub >> authorized_keys
[admin@pc1 ~]$ rm /tmp/id_rsa.pub
[admin@pc1 ~]$ exit
```

If you don't understand what this meant *please* ask an instructor to explain and give you a hand.

OK, so now your public key is sitting in the file /home/admin/.ssh/authorized_keys in the admin account on your neighbor's machine. So, now let's try connecting to admin on your neighbor's machine:

```
$ ssh admin@pc1
```

You should now see something like:

```
$ ssh admin@pc1
Enter passphrase for RSA key 'admin@pc1':
```

And, at this point you type in the *passphrase* you used when creating your public/private key pair on your machine for your account - *not* the password for the admin account on your neighbor's machine.

If you think about this that's pretty neat! Anywhere your public key resides you can log in using one passphrase, and it won't expire.

Now be sure that you log out of your neighbor's machine:

```
[admin@pcN ~]$ exit
```

[Return to Top]

Hervey Allen

Last modified: Wed Nov 28 12:01:44 EET 2007