Adding SSH Keys with CAT

Here's a quick and dirty way to upload SSH keys to a remote user account, using the CAT command.

This can be done, if your local host doesn't have the ssh-copy-id utility.

If you do not have `ssh-copy-id` available, but you have password-based SSH access to an account on your server, you can upload your keys using a conventional SSH method.

We can do this by using the `cat` command to read the contents of the public SSH key on our local computer and piping that through an SSH connection to the remote server.

On the other side, we can make sure that the `~/.ssh` directory exists and has the correct permissions under the account we're using.

We can then output the content we piped over into a file called `authorized_keys` within this directory. We'll use the `>>` redirect symbol to append the content instead of overwriting it. This will let us add keys without destroying previously added keys.

The full command looks like this:

```
cat ~/.ssh/id_rsa.pub | ssh username@remote_host "mkdir -p ~/.ssh && touch ~/.ssh/authorized_keys && chmod -R go= ~/.ssh && cat >> ~/.ssh/authorized_keys"
```

You may see the following message:

```
OutputThe authenticity of host '203.0.113.1 (203.0.113.1)' can't be established.
ECDSA key fingerprint is fd:fd:d4:f9:77:fe:73:84:e1:55:00:ad:d6:6d:22:fe.
Are you sure you want to continue connecting (yes/no)? yes
```

This means that your local computer does not recognize the remote host. This will happen the first time you connect to a new host. Type `yes` and press `ENTER` to continue.

Afterwards, you should be prompted to enter the remote user account password:

```
Outputusername@203.0.113.1's password:
```

After entering your password, the content of your `id_rsa.pub` key will be copied to the end of the `authorized_keys` file of the remote user's account.

You can then, attempt to use SSH key authentication.

Revision #1
Created 1 September 2025 07:49:17 by glwhite
Updated 1 September 2025 07:51:59 by glwhite