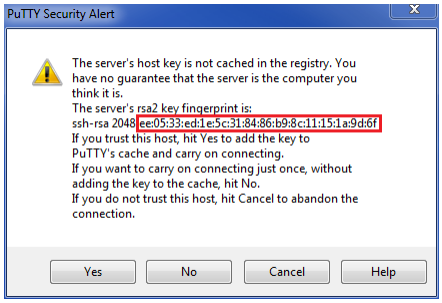
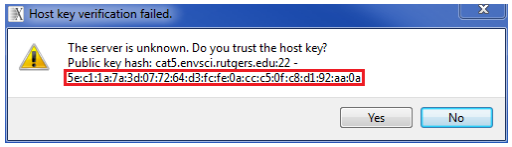
# Cat5 Host Key Fingerprints

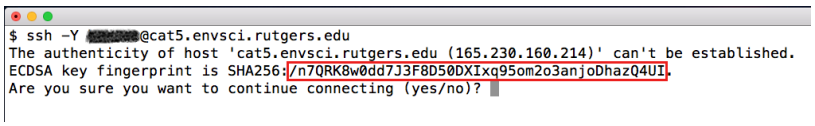
Ensure the computer you connect to is the one you think it is!

## Host Key Fingerprints for Cat5

When you connect to Cat5 for the first time with your SSH or X2Go client, the program won't recognize Cat5's "host key" and will ask if you trust that you are connecting to the right computer. For example:




To answer this question, compare the provided "fingerprint" or "hash" (outlined in red in the examples above) to the valid values in this table:

| Key Type | FP Format | Fingerprint / Hash Value | Shown by Client(s) |
|---|---|---|---|
| ED25519 | SHA256 | arMbmUfQI7livx/EmZT4Qyz85Pz3ND0JeMBhaNAH9x8 | |
| | SHA1 | 5e:c1:1a:7a:3d:07:72:64:d3:fc:fe:0a:cc:c5:0f:c8:d1:92:aa:0a | X2Go |
| | MD5 | 55:87:cd:ef:80:dc:9d:e8:1d:14:87:27:40:00:01:4a | X2Go |
| | Bubble Babble | xilis-cukal-pazob-losak-gygaz-sizab-pufus-hufas-mugun-dipub-paxax | |
| ECDSA | SHA256 | /n7QRK8w0dd7J3F8D50DXIxq95om2o3anjoDhazQ4UI | SSH (Mac/Linux) |
| | SHA1 | 76:83:c5:34:c6:fa:70:eb:11:e9:71:3a:f4:17:40:70:f7:cb:42:f1 | |
| | MD5 | 25:ea:b6:1d:75:17:9e:ea:48:61:50:09:4d:80:57:17 | SSH (Mac/Linux) |
| | Bubble Babble | xitim-fycyf-gucuz-pusev-riguv-nosuf-putic-lobyl-botos-rebaz-cexyx | |
| RSA | SHA256 | rI7gCosqm2qCOkfjS5xWNI+PCZQuasvphNq28NIR2Dg | |
| | SHA1 | 8b:a4:c0:60:77:c6:cc:d7:b9:99:39:7e:90:ff:1b:4b:c5:91:02:0c | |
| | MD5 | ee:05:33:ed:1e:5c:31:84:86:b9:8c:11:15:1a:9d:6f | PuTTY, SSH (Mac/Linux) |
| | Bubble Babble | xodop-gybuk-bytas-kufut-laven-nuvyl-vegiz-zukyg-rycun-cubib-suxex | SSH Secure Shell (Windows) |

If the fingerprint ***exactly*** matches one of these, then choose "yes" to connect. If it does not match any of them, then there is a problem. **Do not connect** and contact help@envsci.rutgers.edu for assistance.

Please see next page for further explanation of why this is necessary.

# Further Explanation

When connecting to a remote computer (e.g. Cat5), you will want to ensure that your connection is secure, so that your password, code, and other information cannot be spied on over the network by bad actors. Programs like **X2Go** and **SSH** clients (e.g. **PuTTY**) assist in this effort by encrypting the data transferred between your local computer and the remote one so it cannot be read by others. (X2Go uses an SSH client underneath to achieve a secure connection.) Even so, the potential still exists that someone could hijack your network connection and insert a third computer in between to collect information as it passes back and forth. This is known as the "man-in-the-middle" attack.

To prevent such an attack, you must always **verify** that the computer you connect to is the one you *intended* to connect to. X2Go and SSH clients do most of that work for you but require your input to complete the process. Every computer has a unique "**host key**" which distinguishes it from other computers you might connect to. (A host key has a secret part that cannot be copied by an imposter.) Once your SSH client (or X2Go) knows the valid host key for a remote computer (e.g. Cat5) it will verify that host key every time you connect. If at some point the key doesn't match, the program will alert you that you may not be connecting to the correct computer. Please pay attention to such alerts!

## Your Role

**The first time** you connect to a remote computer (e.g. Cat5), the SSH/X2Go client doesn't know the correct host key to verify against. Your job is to verify that the host key matches the correct one for the remote computer you intend to connect to. The table above provides the correct keys for Cat5.

A complicating factor is that the host key is too long to practically compare, so a shorter "fingerprint" (also called "hash") of the key is shown instead. Various SSH clients calculate the fingerprint differently, resulting in four possible fingerprint formats for a given host key. In addition, Cat5 has three different host keys for compatibility reasons, and only one of those keys will be chosen for verification by your SSH/X2Go client. That makes *twelve* possible fingerprints to compare against. All of them are correct, but your SSH/X2go client will only pick one to show you. Luckily, you only need to do this process once per remote computer! (Maybe twice if you use both X2Go and SSH.)