# SusSSH: Passphrase phishing via the SSH prompt

Date [ Jun 14, 2019 ]

## The Attack

One of your “friends” tells you to connect to their SSH server, saying they already added your public key. You connect and enter your passphrase.

Nothing super unusual about this process, but with this attack, your “friend” can discover your SSH passphrase.

Try it out yourself by SSHing to

```
sus_ssh.insinger.me
```

(don’t use your real passphrase, intended for OS X)

## How does it work?

This attack is executed by creating an SSH server that uses `none` authentication. This allows any client to create an SSH channel without having to authenticate an SSH key/give a password. Then the server can emulate the client passphrase prompt. The PoC is specifically tailored to OS X’s SSH implementation, but does respond to the SSH username.

The PoC code can be found here. This is a simple PoC but more advanced versions could include:

- dynamic prompt generation based on the client version string (different prompt for PuTTY)
- adding a key cursor (as OS X and some Linux distros do) via iTerm2 Images
- SSH key type identification[1] so I don’t have to guess that most readers use Ed25519

## Is it useful?

The usefulness of this attack is unclear, largely becuase of host key

verification.

To phish someones passphrase you need to: (pick at least 1)

- convince them to connect to a server they've never connected to before that allegedly has their public key already
- convince them to bypass a host key warning
- own the host key of a machine they already connect to (either by owning just the key and MITMing or owning the entire machine)

If you are able to execute either of the last two bullets, you beat host key verification but you might not care that much about phishing an SSH key passphrase. Perhaps the readers can be more creative than I was in thinking of hypothetical uses for this technique.

---

1. I was unable to discern the client SSH key type (RSA, Ed25519, etc) via the connection, as offering `none` authentication prevented me from receiving keys. I could be mistaken about this, but even so a workaround is making the user reconnect due to a fake error (recording their key type the first time). [return]