The sys admin's daily grind: Single-packet authentication

# KEY EXPERIENCE

Conventional, woodpecker-style port knocking is open to sniffing and brute force knocking attacks. Sending an encrypted packet with an access request to the server is safer and more modern. Learn more about Firewall Knock Operator, a.k.a. Fwknop.

**BY CHARLY KÜHNAST**

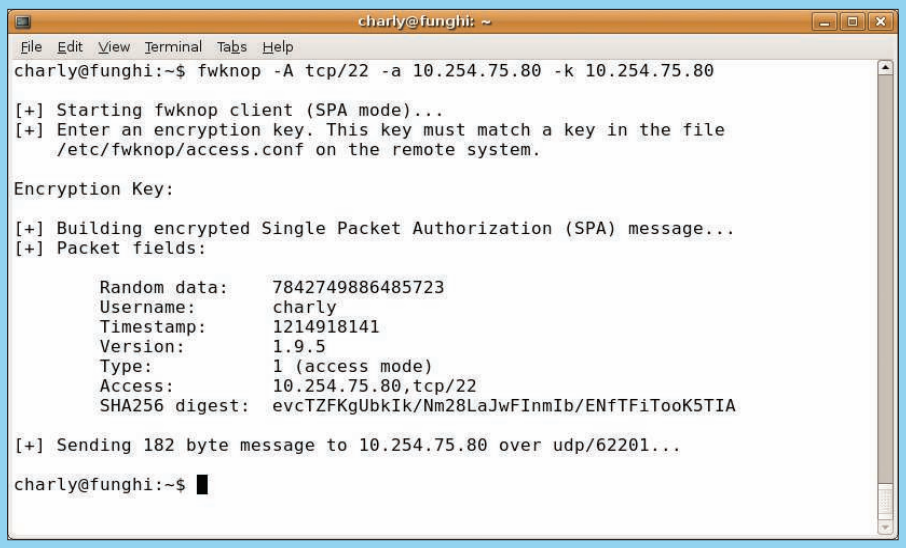


**Figure 1: The client knocking on the door of port 22 is allowed to pass because it possesses the right key.**

Conventional port knocking, which I described last month [1], protects you against attackers who routinely scan whole networks looking for "low-hanging fruit." A cracker who takes more time and logs communications can also identify knocking signals because the sequences will repeat.

In theory, you might consider using lists of one-off knocking signals that become obsolete after use. Unfortunately, this is really complex. Besides, if the administrator is not creative enough, an attacker could just try out popular knocking sequences (port 7000, 8000, 9000, …) to gain access.

Single-Packet Authentication (SPA) is one possible solution. The knocking system sends a single packet containing the encrypted authentication credentials – typically a pass phrase – and the client request to open a specific port. An SPA implementation that works really well is Firewall Knock Operator, or Fwknop [2].



Besides the normal build tools, the installation requires Perl, the libpcap-dev package, and the CPAN *Net::Pcap* module. After installing all of these resources, installing Fwknop is a breeze thanks to the Perl-based installer.

## Matching Knobs

Fwknop comprises the *fwknopd* server and the *fwknop* client. By editing two files below */etc/fwknop/*, you can configure the server; *fwknop.conf* contains the basic configuration. Initially, you will just need to change a couple of parameters, which are tagged *__CHANGEME__*.

The other knobs you could tweak here have very sensible defaults. Note that you need to synchronize the time between the server and the client because if the difference is too big, *fwknopd* will ignore the knocking client.

The entries in */etc/fwknop/access.conf* define how *fwknopd* responds to a client knocking. The secret key that the client uses to identify itself is stored here. The *SOURCE* line can be used to restrict the networks from which the daemon accepts knocking. To set the port that the system opens on successful knocking – for example, *tcp/22* for SSH – you can use *OPEN_PORTS*. Figure 1 shows a successful attempt. The *fwknop* client picks up the key from its own */etc/fwknop/access.conf*.

If the SSH connection doesn't open quickly enough, the *FW_ACCESS_TIMEOUT* on the server triggers. This time is normally set to 30 seconds, but I went for twice that – never rush an admin on the job! ■

**THE AUTHOR**

Charly Kühnast is a Unix operating system administrator at the Data Center in Moers, Germany. His tasks include firewall and DMZ security and availability. He divides his leisure time into hot, wet, and eastern sectors, where he enjoys cooking, fresh water aquariums, and learning Japanese, respectively.

**INFO**

[1] "Knock-Knock" by Charly Kühnast, *Linux Magazine*, September 2008, *http://www.linux-magazine.com/issues/2008/94/knock_knock*

[2] Fwknop: *http://www.cipherdyne.org/fwknop/*