Nr. 06

Was ist Tunneling?

Im Zusammenhang mit unserem Sicherheitsprojekt "Transistor" kommt dem Tunneling oder Port Forwarding eine wichtige Rolle zu. Dabei erlaubt Tunneling, Daten beliebiger Internetdienste innerhalb des SSH Protokolls zu übertragen. Der Vorteil liegt dabei im Senden von unverschlüsselten Daten über einen sicheren Träger, eben den SSH Tunnel. Um dies zu veranschaulichen, wollen wir zuerst eine klassische Verbindung ohne Tunneling betrachten:

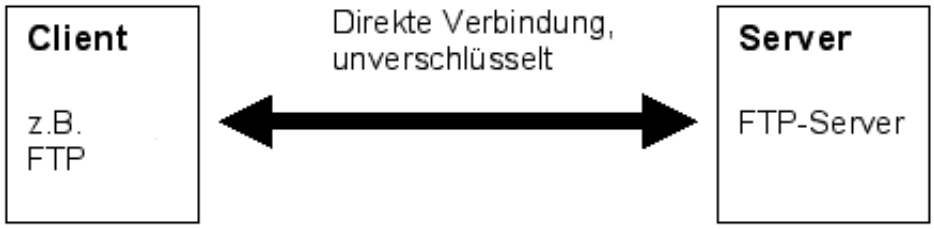

Abb. 1: Klassische FTP-Verbindung ohne Tunneling

Der Benutzer wählt über den Client z.B. FTP den Server an, um Files zu transferieren; dabei gehen alle Daten, inkl. Passwort im Klartext über das Netz. Wenn wir beispielsweise mit Fetch auf den FTP-Server des ZI zugreifen, wird Fetch eine Verbindung über den standardisierten Port für FTP herstellen, d.h. Port 21. Das ist sozusagen der Kanal im Netz, den FTP erhält, um sich gegen andere Dienste abzugrenzen und eine Verbindung aufzubauen. Danach legen der FTP-Server und der FTP-Client in der Regel fest, auf welchem Port die Verbindung für den Datenaustausch erfolgt.

FTP via Secure Shell

Es besteht nun die Möglichkeit, über das Port-Forwarding den FTP Verkehr durch einen anderen Port zu schicken, den SSH Port (vgl. Abb. 2) Dazu muss aber FTP gezwungen werden, die Daten über den Standard-Port 21 zu schicken. Clients, welche das können, schalten dazu den Passiv-Modus ein.

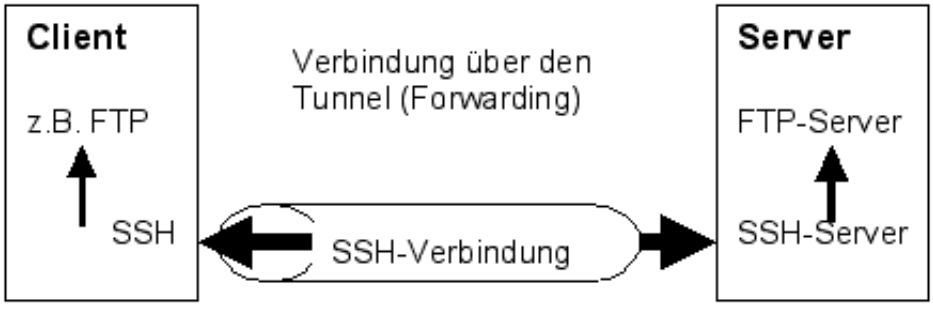

Abb. 2: FTP via Secure Shell

Gemäss dem Schema passiert folgendes: der Benutzer hat auf seinem Computer ein FTP-Programm, dieses schickt seine Signale an den FTP-Port 21 desselben Rechners. SSH ist so eingestellt, dass es auf diesen Port hört, die Signale aufnimmt und über die eigene sichere Verbindung an den entfernten Rechner schickt, wo wieder SSH installiert ist. Dieses schickt dann die Signale nochmals weiter an den dortigen FTP-Server. Ein Beispiel für ein solches Port Forwarding finden Sie hier.