SSH statt VPN

Dokumenthistorie:
10.12.2007: Erstellung
14.12.2007: Nachtrag SSH-Zugriff

Vorbemerkung:
In meinem VPN-Tutorial verweise ich unter anderem auf Microsoft-Links für den Download von Service Packs und Administrator-Toolkits. Diese Links sind nicht mehr gültig, da Microsoft für den Download dieser Dateien mittlerweile eine Gültigkeitsprüfung vorschiebt. Ohne bestandene Gültigkeitsprüfung geht also leider kein direkter Download mehr.
Dieses Tutorial entstand, weil die Einrichtung und der Betrieb von VPNs aus verschiedenen Gründen an seine Grenzen kam:

- Unter der Verwendung bestimmter mobiler Internetzugänge erhält man unter Umständen keine IP aus dem öffentlichen Netz, damit scheidet der Aufbau eines Roadwarrior-VPNs aus.
- Auf meiner Linuxkiste bin ich immer noch nicht dazu gekommen, die VPN-Sache einzurichten, ssh und vnc sind aber drauf. Damit wäre ich unabhängiger von meiner Windowsbox, um mal eben jemanden auf der Arbeit von zu Hause aus beizustehen, weil die Linuxkiste eigentlich immer an ist, die Windowsbox aber nicht.

Sehr hilfreich fand ich in diesem Zusammenhang Eckis ssh-lan Anleitung und die Dokumentation zu putty (einem freien SSH-Client für Windows), erhältlich unter
http://www.chiark.greenend.org.uk/~sgtatham/putty/
oder nach putty suchen... . Zum tieferen Verständnis ist die Lektüre der beiden Anleitungen sehr zu empfehlen.

Voraussetzungen:
Diese Anleitung setzt voraus, dass ein Roadwarrior entweder auf einen Server oder einen Client in einem Netzwerk zugreifen möchte. Einstiegspunkt ist bsw. ein IpCop-Router, muss aber nicht sein, letzten Endes muss der Router via DynDNS aus dem Internet erreichbar sein und einen Port auf einen anderen Rechner weiterleiten können (dieses muss also eingerichtet sein). Ich fahre bsw. auch Konstellationen, bei denen eine normale DSL-Router-Modem Kombination zum Einsatz kommt, die einen Highport auf einen existierenden Rechner weiterleitet.
Der Einfachheit halber folgendes Szenario (alle Adressen und Ip-Nummern sind fiktiv und durch eigene Angaben zu ersetzen.):
Einstiegspunkt ist ein IpCop-Router. Bei no-ip.com habe ich ihn registriert und kann ihn unter der Adresse mein-Router.no-ip.org erreichen.
Auf dem Router selbst trage ich eine Portweiterleitung ein (Registerreiter Firewall, Portweiterleitung). Protokoll ist TCP, Quellport ist ein beliebiger freier Highport (bsw. 55555). Ziel-IP ist die IP des zu erreichenden Servers, Zielport ist in dem Fall der ssh-port meines Servers. (habe ich einen einzelnen Rechner hinter einem Modemrouter, der eine feste Adresse hat, könnte ich theoretisch diese Adresse eintragen... .).
Natürlich muss ebenfalls der ssh-Zugriff erlaubt sein und die tcp-Weiterleitung zugelassen werden (Registerreiter System, ssh-zugriff, und dort die relevanten Einträge vornehmen (sonst wäre das ganze auch witzlos).

Los geht's:
Putty herunterladen und in einem Verzeichnis der eigenen Wahl ablegen. Es handelt sich um eine einzelne ausführbare Datei, die einfach per Doppelklick gestartet wird. (unter Windows.).
Für das Ausfüllen der einzelnen Felder ist die Dokumentation von Simon Tatham eigentlich so vollständig, dass es hiesse, Eulen nach Athen zu tragen, wenn man diese Beschreibung wiederholen

wollte.
Viel interessanter finde ich die Möglichkeit, per Kommandozeile zu arbeiten (geht nämlich auch!). Um eine Verbindung zu meinem Server aufzubauen, gebe ich in der Windows Kommandozeile folgendes ein:
Pfad-zu-Putty\putty mein-Router.no-ip.org -P 55555
Wir erinnern uns: mein-Router.no-ip.org ist der Name der Adresse, unter der mein Router im Internet erreichbar ist. Das grosse P signalisiert, dass ich auf einen bestimmten Port verbinden möchte und die 5 mal die 5 spezifiziert genau diesen Port. Der Router leitet die Anfrage prompt weiter, so dass ich einen Login-Prompt auf dem Zielrechner (Server) erhalte.
Interessant wird es, wenn ich bestimmte Informationen gleich mit auf den Weg gebe: -l spezifiziert den Login-Namen, -pw spezifiziert das Kennwort. Um einen bestimmten lokalen Port auf meinem Rechner mit einem bestimmten Zielrechner und Port im Netzwerk hinter der Firewall und hinter dem Server zu erreichen, um bsw. vermittels UltraVNC eine Fernwartung durchzuführen, kann ich das wie folgt erreichen: -L 6000:ZielIP:Zielport (bei UltraVNC ist das 5900, auf dem Zielrechner muss der UltraVNC-Server für die lokale Firewall (bsw. WindowsXP) freigeschaltet sein.).
In ganz lang also:
Pfad-zu-Putty\putty -l meinlogin -pw meinpasswort -L 6000:ZielIP:5900 mein-Router.no-ip.org -P 55555
und schwupps, kann ich mein lokales UltraVNC starten und eine Verbindung auf localhost:6000 starten und den gegnerischen Rechner fernwarten.
Statt der Übergabe eines Passworts könnte man versuchen, mit ssh-Schlüsseln zu arbeiten, ich habe dies auf einigen Windowskisten hingekriegt, auf anderen nicht (ohne herauszufinden, warum).

Auf meiner Linux-Kiste ist ssh installiert, dort öffne ich also ein Terminalfenster und gebe folgendes ein:
ssh -l meinlogin -L 6000:ZielIP:5900 mein-Router.no-ip.org -p 55555
Beachte den Unterschied: keine Angabe von Passwort möglich (dafür funktioniert der Schlüsselaustausch unproblematisch) und die Angabe des Ports wird mit einem kleinen p vorgenommen statt mit einem grossen P. Ansonsten kann ich auch hier anschliessend mit einem VNC-Viewer eine Verbindung zu localhost:6000 aufbauen und den gegnerischen Rechner fernwarten.
Geht schneller als mit VPN, funktioniert auch mit Einwahlverbindungen, die keine öffentliche IP vergeben (bsw. BASE-Internet-Flatrate) und ist schön schlank. Rein theoretisch könnte man sogar mit einem ssh-Client für ein mobiles Gerät das gleiche erreichen (wenn nur die Schrift nicht so klein wäre...).

Ralf Petry.
Berlin, 10.12.2007
ralf.petry@cityweb.de