Dabei wird davon ausgegangen, daß innerhalb von 20 Sekunden die Nagios-Arbeit auch wieder abgeschlossen ist, denn danach baut sich auch die SSH wieder ab.

Und schon können Sie Ihr Monitoring verwenden!

OpenSSH bietet eine Menge an Möglichkeiten. Die Herausforderung besteht darin, diese Fähigkeiten in Zusammenhang mit einer konkreten Problemstellung zu bringen. Die nächsten Beispiele zeigen weitere Möglichkeiten.

8.5.2 NX/FreeNX und OpenSSH

Einer der wohl bekanntesten freien Remote Desktops ist *VNC* (`http://www.realvnc.com/`). Es gibt noch weitere Implementierungen, wie zum Beispiel *TightVNC*.

VNC ermöglicht die Fernwartung von Desktops unter Linux, Windows oder Mac OS. Die freie VNC-Version verfügt jedoch über keine Verschlüsselung, so daß eingegebene Kommandos leicht abgehört werden können. Dieses ist ein sicherheitskritisches Problem, vor allem wenn man bedenkt, daß es sich hierbei um eine Remote Desktop-Implementation handelt.

Mit Hilfe von NX der Firma NoMachine (`http://www.nomachine.com`) bzw. der freien Implementation FreeNX (`freenx.berlios.de`) steht aber eine Lösung für dieses Problem bereit, bei der SSH eine tragende Rolle spielt. NX/FreeNX können für denselben Zweck wie VNC eingesetzt werden, dabei ist aber eine Verschlüsselung der Datenübertragung mit SSH bereits integriert.

Gegenüber VNC bieten NX/FreeNX außerdem noch weitere Vorteile:

- Datenkompression des Netzwerkverkehrs
- Reduzierung der Roundtrips zwischen X-Client und X-Server
- Anlegen eines Caches für schon übertragene Daten

NX/FreeNX basieren auf dem Client/Server-Prinzip. NX/FreeNX-Clients existieren für Windows, Linux, Solaris, Mac OS X, Zaurus, iPAQ, X-Box und Playstation 2. NX/FreeNX-Server gibt es für Linux und Solaris, und sie sind in der Lage, NX-Sessions an VNC-Server oder Windows-Terminalserver (per RDP) weiterzuleiten und diese nochmals zu komprimieren. In der höchsten Kompressionsstufe (für langsame Modemverbindungen) erreicht NX eine effektive Kompression von ca. 70:1. So ist es selbst mit einer ISDN-Verbindung problemlos möglich, mit einer graphischen Benutzeroberfläche, wie KDE, flüssig zu arbeiten.

Für KDE existiert die Clientapplikation kNX, die aber auch unter Gnome verwendet werden kann. Abbildung 8.16 zeigt kNX mit den Konfigurationseinstellungen. Für eine erste Anmeldung müssen Sie hier zunächst eine neue Verbindung einrichten. Dazu sind die folgenden Schritte notwendig:

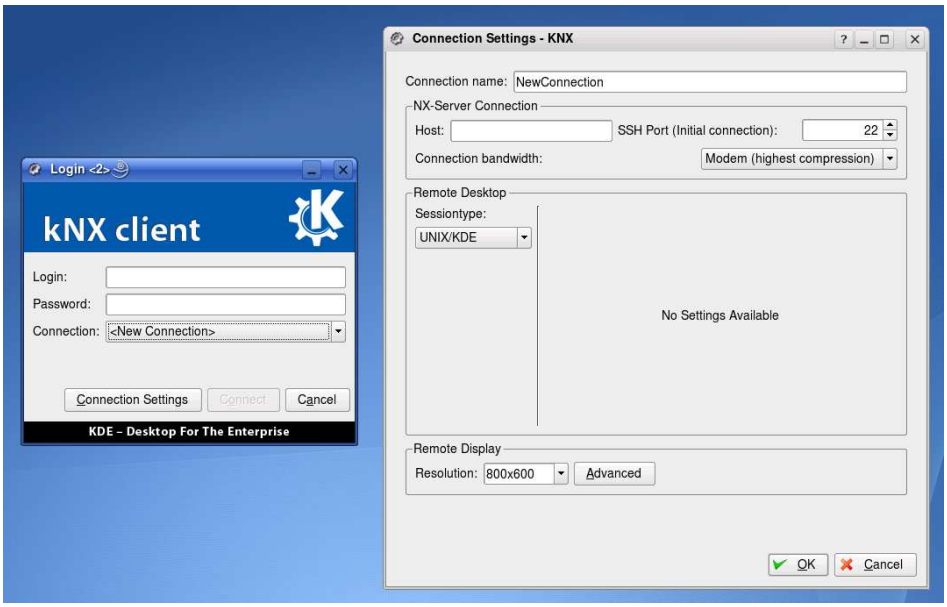

Abbildung 8.16: kNX-Client mit Konfigurationseinstellungen

- ❏ Starten Sie kNX
- ❏ Klicken Sie im Fenster „kNX client“ (Abbildung 8.16 links) auf „<New Connection>“ (Verbindungseinstellungen).
- ❏ Geben Sie im neu auftauchenden Fenster (Abbildung 8.16 rechts) einen Namen für die Verbindung ein, beispielsweise den Servernamen.
- ❏ Geben Sie Hostinformationen, die Portnummer und die Bandbreite
  - ➢ Modem (highest compression)
  - ➢ ISDN
  - ➢ ADSL
  - ➢ WAN
  - ➢ LAN (least compression)

  für Ihre Verbindung ein.
- ❏ Wenn Sie KDE verwenden, wählen Sie den Sessiontype UNIX/KDE
- ❏ Wählen Sie eine Bildschirmauflösung
  - ➢ 800x600
  - ➢ 1024x768
  - ➢ Fullscreen
- ❏ Klicken Sie auf OK.

- Wählen Sie nun im Fenster „kNX client“ (Abbildung 8.16 links) unter „Connection“ die gerade konfigurierte Verbindung auf den entfernten Rechner.
- Geben Sie Ihr Login und Ihr Paßwort für diesen entfernten Rechner ein.
- Klicken Sie auf Connect

Sobald die Verbindung besteht und die Fernverbindung auf Ihrem Bildschirm sichtbar ist, können Sie auf die entfernten Anwendungen genauso zugreifen und den entfernten Computer genauso nutzen, als würden Sie direkt vor ihm sitzen. Wenn Sie die Option Fullscreen gewählt haben, können Sie mit der Tastenkombination Strg+Alt+F die Ansicht etwas verkleinern, so daß Sie unter KDE auf die KDE-Menüleiste zugreifen oder auch das Fenster der Fernverbindung minimieren können.

Für alle großen Linux-Distributionen existieren fertige Pakete für NX/FreeNX, so daß sich die Installation üblicherweise auf das Einspielen der Pakete und die Konfiguration von OpenSSH beschränkt.

Unter SUSE LINUX sind beispielsweise Server-seitig die RPM-Pakete NX bzw. FreeNX zu installieren. Auf den Clients ist eine Installation der RPM-Pakete NX, kNX bzw. nxclient (für Nicht-KDE/Gnome-Sitzungen) notwendig.

Je nach Distribution muß ggf. der NX-Server noch eingerichtet werden, indem man den folgenden Befehl als `root` eingibt:

```
nxsetup --install --clean --purge --setup-nomachine-key
```

Damit NX/FreeNX seine Arbeit verrichten kann, müssen auf den Firewalls, über die der Datenverkehr geht, die Ports für SSH, üblicherweise 22, 5000 bis 5009 sowie 7000 bis 7009 geöffnet sein.

Dieser kurze Einblick soll an dieser Stelle genügen. Die Anwendungsmöglichkeiten von NX/FreeNX im Zusammenspiel mit SSH sind so vielfältig, daß eine weitere genauere Darstellung den Rahmen dieses Buches sprengen würde. Weitere Informationen zu Installation und Konfiguration findet man auf den oben genannten Webseiten von NX und FreeNX.

8.5.3 CVS mit OpenSSH

CVS (Concurrent Version System) ist der Klassiker unter den freien Systemen zur Versionskontrolle. Viele Versionen, zumeist die älteren, verwenden `rsh`, um sich mit den Repositories auf einem Remote Host zu verbinden. Das ist offensichtlich eine sehr unsichere Variante, da die Daten unverschlüsselt zwischen Client und Server ausgetauscht werden. Es bietet sich also an, CVS zusammen mit OpenSSH zu verwenden. Das geschieht denkbar einfach mit Hilfe der Umgebungsvariable `$CVS_RSH`.