Firewall-Administration in einem größeren Netzwerk

GUUG-Frühjahrsfachgespräch 2008

Detlef Lannert

Zentrum für Informations- und Medientechnologie der
Heinrich-Heine-Universität Düsseldorf

München, 14. 03. 2008

Wer bin ich?

- Detlef Lannert
- Dipl.-Mathematiker
- wiss. Beschäftigter beim Zentrum für Informations- und Medientechnologie der Heinrich-Heine-Universität Düsseldorf
  (d. h., beim früheren Universitätsrechenzentrum)
- Schwerpunkte: Netzwerke, IT-Sicherheit, Linux
- Hobby-Missionar für Linux, Python, freie Software; Koautor „Linux im Netzwerk“ (A-W)

Ein größeres Netzwerk??

Das ist etwas geprahlt – für uns heißt das:

- Universitätsnetzwerk (Campus-Uni)
- früher mal mehr als 20 000 Studierende
- 3 größere und 2 kleinere Fakultäten
- mehr als 7 000 Endsysteme
- ca. 40 „eigenständige“ Bereiche mit separaten Subnetzen und VLANs
- durchgehendes Backbone, jetzt 10 GBit
- zwei separate Leitungen zum Provider (je 100 MBit)

Also: „größer“ == zu groß für reine Handarbeit

Warum Firewalling?

- Die Beschwerden
  - unserer Nutzer über Angriffe von außen
  - externer Internet-Bewohner über unsere Nutzer
  - interner Nutzer über Angriffe aus anderen Instituten bzw. Fakultäten

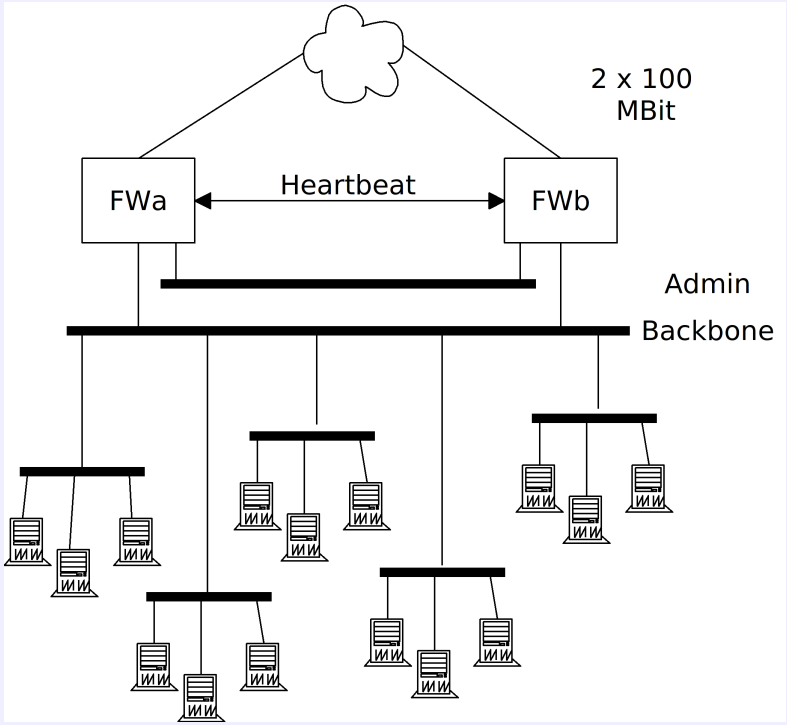
  nahmen exponentiell zu;
- ebenso der Zeitaufwand, um diesen nachzugehen.
- Externe Ko-Administration, PFUH (Populäre Fileserver mit Daten unbekannter Herkunft), Hausbesuch der Kripo, ...
- Innerhalb der Bereiche sorgen lokale Firewalls und die Autorität der Chefs (oder der Stärkeren) meist für Frieden $\Rightarrow$ Filterung an Bereichsgrenzen angesagt
- Firewalling ist einfach “state of the art”

Warum (eigene) Linux-Lösung?

- Linux-Netfilter ist leistungsfähig, flexibel, vielseitig
- Übersichtliche Konfiguration durch Unterprogrammtechnik für die Firewall-Regeln
- Keine VLAN-Beschränkungen
- IP-Routing (Kernel), Routing-Protokolle (Quagga) – ja, auf der Firewall!
- Weitgehende Debugging- und Analysemöglichkeiten
- Knowhow (und Hingabe) waren vorhanden

Netzstruktur

Ein etwas vereinfachter Blick auf das HHU-Netz:



Erste Schritte

Wir haben klein angefangen:

- 2 Transtec-Rackmount-PCs
- zuerst nur Inhouse-Netz (und das nur teilweise)
- Routing weiterhin durch zwei Cisco-Router
- Failover der Gateway- und Routing-Adressen durch Heartbeat (active-active)
- Inbound- und Outbound-Filterung!

Spannende Fragen:

- Läuft das stabil genug?
- Klappt das Clustering?
- Leistungsfähig genug?
- Bleibt es handhabbar?

Erste Erfahrungen

- Stabilität: OK, auch im Dauerbetrieb
  zeitweise allerdings Probleme mit GF-Kartentreiber;
  Serverausfälle zum Ende der Garantiezeit
- Clustering: OK, aber Gefahr von Split-Brain-Situationen;
  eigene Resource-Prozeduren sinnvoll
- Leistungsfähigkeit: OK
  aber einige sysctl-Parameter verändern
- Handhabbarkeit: OK – mit geeigneter Software-Unterstützung

Akzeptanzfragen

- Scheibchenweise Ausweitung auf die anderen Subnetze
- Vorher mit den Anwendern sprechen
- Feste Ansprechpartner / Koordinatoren benennen lassen
- Änderungswünsche über diese kanalisieren
- Beratungsmöglichkeiten
  („Ihr wollt das nicht mit FTP machen!“)
- Nur wenige Dienste (Ports) werden tatsächlich benötigt
- Die meisten Anwender merken nichts (Nachteiliges)
- Nur einzelne Poweruser „brauchen“ weitreichende Freigaben
- Viel Verständnis und Rückhalt bei den Institutsleiter(inne)n, Dekanen etc.

Veränderungen

- Neue Hardware: 2 SunFire X4200, je 2 Dual-Core-AMD 2,2 GHz, 4 GByte
- Separates Administrationsnetz (2 kleine Server als HA-Cluster)
- Weiterentwickelte Programme / Prozeduren zur Administration
- Logging remote über Syslog-NG
- Routing auf die Firewalls übertragen, OSPF mit Provider
- Nebenjobs für die Firewalls: DHCP-Server, MAC-Erfassung (einige andere konnten verhindert werden)

Neuere Erfahrungen

- Hardware läuft stabil (ist noch in der Garantiezeit! ;) – auch die Netzwerkkarten
- CPU-Auslastung gering (unter 10 % bei 500 MBit/s)
- Latenz nicht spürbar
- Zahlreiche Ausfälle (z. B. einer Außenleitung) blieben praktisch unbemerkt
- Einige Anwendungen sind filterfeindlich und schlecht dokumentiert
- Änderungshäufigkeit für den Regelsatz: ca. 150/Jahr
- Ansonsten läuft das so vor sich hin
- Debian “stable” hat sich gut bewährt

Verkehrsregeln

- Die iptables-Kommandos will man nicht von Hand eintippen
- Andererseits sind komplexe Abhängigkeiten von (fremder oder anfälliger) Software zu vermeiden „Ich kann Ihnen das erst freischalten, wenn ich eine neue Version von XYZ habe.“
- Erstmal naheliegend: Shell-Prozeduren
- Schnell kommen einige ausgefuchste Shell-Funktionen hinzu
- Irgendwann skalieren Geschwindigkeit, Programmier- und Pflegeaufwand nicht mehr
- Dann erfindet man halt einen neuen Regelgenerator (mit GUI? Nein, mit einer neuen Makrosprache)

Beispiel: Shell-Prozedur

Zugriff mit NTP und Time auf Zeitserver erlauben:

```
for server in 134.99.128.79 134.99.128.80; do
    for proto in tcp udp; do
        for port in 37 123; do
            iptables -A $chain -d $server \
                     -p $proto -dport $port \
                     -j ACCEPT
        done
    done
done
```

Durch Zusatzmodule wie `multiport` wird es letztlich auch nicht einfacher!

Firewall-Regeln 3.0

- Regelgenerierung jetzt auf den Admin-Rechnern
- Angepaßte Makrosprache: Möglichst übersichtlich, nicht geschwätzig, listenorientiert, mächtig
- Source-Verwaltung mit VCS (Mercurial)
- Interpreter in Python implementiert (mit Pyparsing)
- Eingabedateien für `iptables-restore -n`
  (sind notfalls editierbar!)
- Anbindung an Datenbanktabellen (z. B. für Subnetze)
- siehe Beispieldateien!

© Detlef Lannert
ZIM
HEINRICH HEINE UNIVERSITÄT DÜSSELDORF

Verwaltungsakte

Shell-Kommandos für administrative Aktionen auf den Firewalls:

```
hhufw status all
hhufw start okklumentik
hhufw showrules -less arithmantik
hhufw loadrules arithmantik
hhufw stop all
```

Schlußbemerkungen (aus Admin-Sicht)

- Schuldzuweisungen an die Firewall bei Netzproblemen sind üblich, aber meist unberechtigt
- Gute Diagnosemöglichkeiten (`arping`, `ip route`, `ip neigh`, `tcpdump`, Netfilter-Counter)
- Gelegentliches kontrolliertes Failover (und auch mal ein Reboot) ist sinnvoll
- Infizierte Rechner fallen auf (im Log); IDS geplant
- Besondere „manuelle“ Rulechains zum Abblocken von Problembären drinnen oder draußen sind nützlich
- Keine Begeisterung für NAT
- Besserer (Tag- und) Nachtschlaf

Ihre Fragen sind willkommen!

Kontakt: `lannert@uni-duesseldorf.de`