0. Wiederholung:

- inetd:
  - konfiguriert in /etc/inetd.conf
  - /etc/services => beschreibt welcher Port für welchen Dienst zuständig ist.
  - inetd überwacht Ports

1. FTP-Deammon als ON-Demand-Server einrichten:

- tcpd:
  - stellt Zugangskontrollmechanismen für andere Programme zur Verfügung
  - Konfigurationsdateien:
    - /etc/hosts.allow: welche Anwendung auf welchen Dienst zugreifen darf
    - /etc/hosts.deny: welche Anwendung auf welchen Dienst **nicht** zugreifen darf
  - Es wird immer zuerst die hosts.allow ausgelesen.
    --> sucht nach Zugangsregelung über aktive Anwendungen
    --> falls gefunden: ausführen
    --> sonst in der hosts.deny suchen

1. Installation von in.ftpd
2. Starten:
   - starten wie normal nicht möglich, da On-Demand-Server(!!!)
   - starten über inetd:
     - inetd starten (wird beim Systemstart nicht mitgestartet)
     - in inetd.conf die Einstellungen ändern => soll in.ftp überwachen
       - Tabellenform von inetd.conf:

ServiceName	SocketTyp	Protokoll	flag(s)	User	ServerPath	ServerArgs
#ftp	stream	tcp (udp)	nowait (wait)	root	/usr/sbin/tcpd	

(--> flag nowait: Socket nach Ende gleich wieder freigeben. Folgende Anfrage mit neuem Prozess bearbeiten)

   - inetd anweisen, dass er bei Anfrage den in.ftp starten soll.
     --> in inetd.conf die Kommentarzeichen der Zeile "ftp stream tcp nowait root ..." (s.o.) löschen

3. Test:
    - inetd restarten/starten
    - "ftp localhost" eingeben

4. Zugriffskontrolle:
    - hosts.allow & hosts.deny:
        -> All : All => alle Dienste für alle
        -> 192.168.1 => alle Rechner in dem Netz
        -> .web.de => alles mit "web.de" am Ende

2. Server:

- telnetd
    - On-Demand-Server
    - in.telnetd
    - Problem: sendet User und Passwort unverschlüsselt

- squid
    - Stand-Alone-Server
    - Standard Web-Proxy unter Linux
    - Funktionen:
        - Cache für Internet-Seiten
        - Zugriffe können gefiltert werden (URLs, Aktionen(*.exe-Dateien)
    - Konfigurations-Datei:
        - /etc/squid/squid.conf
        - anzeigen mit "cat squid.conf | grep -v "#""
            => Anzeigen ohne auskommentrierte Zeilen
        - # ACCESS CONTROLS
            - Tag acl (access control list)
            --> definieren Variablen
            --> legen die spezifischen Kontroll-Variablen fest
            --> Struktur der Definition:
                acl Name des Parameters Quelle/Ziel IP
                acl all src 0.0.0.0/0.0.0.0
                --> alles was von der Quelle (src) mit beliebiger IP kommt
                --> mögliche Quell/Zielangaben:
                    src
                    dst Destination

srcdomain	Domain-Name
srcdom_regex	Domain-Name wobei reg. Ausdr. mögl.
url_regex	gesamte URL + Pfad (www.heise.de/public)
urlpath_regex	URL-Pathangabe (/public)

- Filterregeln: Tag http_access
    - Default-Wert: deny all
        => gemeint ist alles (da all) mit der Quell-IP, die in "acl all src 0.0.0.0/0.0.0.0" definiert wird
- /var/log/squid/access.log
    - protokolliert die Zugriffe
    - "tail -f /var/log/..."
        -> zeigt die Meldungen immer aktualisiert an
    - Zeile "TCP_MISS"
        - im Cache nichts gefunden -> Original abrufen
    - Zeile "TCP_MEM_HIT"
        - Seite im Cache gefunden -> muss nicht vom Server geholt werden.