

## Kóstoljunk bele a tisztán linuxos irodába!

Néhány egyszerű, elegáns és gazdaságos módja annak, hogy munkaállomásainkat Linuxra állítsuk át.

**F**rançois, csodálatos látványt nyújt ma az étterem! Menj, kérlek, még egyszer körbe, hogy minden munkaállomás rendelkezze-e a szükséges eszközökkel – szeretném, ha minden tökéletesen működne.

Bonsoir, mes amis, isten hozott Chez Marcelnél, a kitűnő Linux-konyha és a csodálatos borok otthonában! Foglaljatok helyet, és helyezétek magatokat kényelembe! François azonnal hozza a bort. Siess! Az 1999-es Chateaufort-du-Pape illeni fog az alkalomhoz.

Azok számára, akik egy ideje látogatják már éttermünket, biztos örömet okoz az a tény, hogy számos vállalat (és magánszemély) keres más választási lehetőséget jelenlegi asztali operációs rendszere helyett. A Linux a kézenfekvő megoldás, és az átálláshoz számos viszonylag fájdalommentes lehetőséget kínál. Sajnos az újabb Linux-rendszer-csomagok a KDE vagy Gnome futtatásával meglehetősen nagy erőforrásigényt támasztanak, sok vállalat pedig éppen a szerényebb PC-ken is működő rendszerek lehetőségét keresi. Szerencsére a Linux lehetőséget ad rá, hogy a munkaállomásokra csak a legszükségesebb összetevők kerüljenek, a munka nagy része pedig a nagy teljesítményű kiszolgálót terhelje.

A telepítésnek ezt a módját nevezik *vékony ügyfél modellnek*. Hogy pontosan milyen „vékony” lehet ez az ügyfél, az a választott megközelítéstől függ. Ha kitűnő alapanyagok széles választéka áll rendelkezésünkre, elvárható, hogy sokféleképpen lehessen elkészíteni a menüt. A friss homár megpárolva, vajöntettel és fokhagymával tálalva nyilvánvalóan kitűnő, de nem ez az egyetlen elkészítési módja. Ugyanez igaz arra, amit Linux-konyháinkban készítünk, ezúttal a vékony ügyfél megvalósítására. Az egyik megközelítés még a számítógép újratelepítését sem igényli.

A vékony ügyfél megvalósításának talán legegyszerűbb módja, ha a munkagép nem futtat ablakkezelőt, hanem azt a kiszolgálóról kapja. Nézzük ennek megoldását:

```
startx /usr/X11R6/bin/xterm
```

A kapott eredmény messze nem mondható érdekesnek. Egy dísztelen, öreg X-felületet kapunk, amiben semmi nem fut, csak az X-terminál, semmi ablakkezelés vagy ilyesmi. Mit lehet ezzel így kezdeni? Egy dolgot biztosan, az OpenSSH segítségével csatlakozni a kiszolgálónkhoz, az alábbi módon:

```
ssh -C -X -l user_name server_name
```

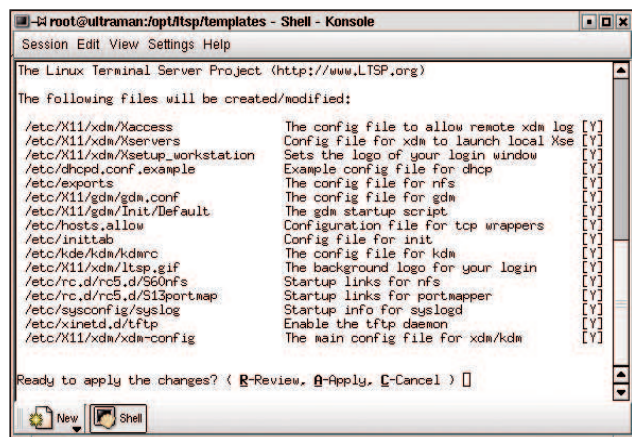
A -C kapcsoló elindítja a biztonságos adatfolyam tömörítését, míg a -X lehetővé teszi az X-kérések továbbítását. Amint a parancssorban találjuk magunkat, már indíthatjuk is az asztalunkat a távoli gépen a startkde vagy wmaker vagy más általunk kedvelt asztal indítóparancsával. Hirtelen egy olyan grafikus asztali környezetben találjuk magunkat, ami a távoli kiszolgálón fut.

Megjegyzendő, hogy a kiszolgáló `/etc/ssh/sshd_config` fájljában

az X11-továbbítást engedélyezni kell, mielőtt a kiszolgáló az ügyfél kéréseit fogadná. A fájl megfelelő kapcsolójának beállítása az alábbi módon történik:

```
X11Forwarding yes
```

Ha meg kell változtatnunk a kapcsolót, az sshd újraindítása is szükséges lesz, amit az `service sshd restart` paranccsal végezhetünk el. A távoli munkavégzés ez a módja remekül működik, és még akkor is igen hasznos, ha már futtatunk egy grafikus Linux-felületet. Leginkább akkor vesszük hasznát, ha nélküle a számítógépteremig kellene kutyagolnunk, ami természetesen az épület másik végében található.



Az LTSP beállítása

Ha a saját felületeket is futtatni szeretnétek a kiszolgálón lévő mellett, a CTRL-ALT-F2 (vagy F3, F4, és így tovább) billentyű-kombinációval kapcsoljatok át valamelyik karakteres konzolra, és indítsátok el egy második X-folyamatot, valahogy így:

```
startx /usr/X11R6/bin/xterm -- :1
```

A -- :1 beállítás (figyeljünk a kettős kötőjelre!) egy második grafikus felületet indít el – egy újabb unalmas xterm ablakkal és egy újabb unalmas háttérrel. Innen az ssh-t használhatjuk a kiszolgálóhoz való csatlakozáshoz, és bármilyen tetszőleges ablakkezelőt elindíthatunk. A grafikus felületek között a CTRL+ALT+F7 CTRL+ALT+F8 lomkiszolgálókkal váltogatunk. A beállítás hátrányai között kell megemlítenem, hogy az a szó szoros értelmében nem tekinthető „vékonynak”, és az OpenSSH-adatfolyam a tömörítés mellett is lassúnak tűnhet. Egy X-folyamat futtatásának (az ügyfélgép minél kisebb terhelése mellett) legjobb módja az XDMCP (X Display Manager Protocol, azaz X-megjelenítéskezelő protokoll) használata. Indításához az ügyfélgép Linux-parancssorába egyszerűen a következő sort gépeljétek be:

```
X -query kiszolgalo_neve
```

És csodák csodája: egy grafikus bejelentkezőképernyőt láthatok, éppen olyat, mintha a kiszolgálón futna. Óh, mes amis, mint azt bizonyára sejtitek, ennek az egyszerű receptnek az elkészítéséhez szükség van egy kis előkészületre, még mielőtt a parancsot futtathatnánk – de nagyon egyszerű. Mialatt François újratölti poharaitokat, el is magyarázom. Ahhoz, hogy a kérést küldő gép számára grafikus bejelentkezőképernyő jelenjen meg, a `/etc/X11/xdm/Xaccess` fájlban engedélyezni kell az alábbi sorokat:

```
*                               #bErmelyik gØp
                               #kaphat bejelentkezi
                               #ablakot
*   CHOOSER BROADCAST          #bErmelyik indirekt gØp
                               #kaphat vElaszt t
```

Továbbá szükség van az `xdm` (vagy a `kdm`, vagy a `gdm`) futására, ennek egyik gyors megoldása, ha a kiszolgálót 5-ös futás-szintre kapcsoljuk. Akik közületek grafikus bejelentkező ablakkal indítják a gépet, valószínűleg átugorhatják a következő néhány sort. Ahhoz, hogy 5-ös futási szinten indítsunk szolgáltatásokat, a `/etc/inittab` fájl `initdefault` kapcsolóját 5-ösre kell állítanunk:

```
id:5:initdefault:
```

Ha a gép nem grafikus felülettel indul, a fenti 5 helyett 3-nak kell állnia. Még egy utolsó beállítást kell végrehajtani ebben a `/etc/inittab` fájlban. Nézzük meg a fájl végét, ahol valami ehhez hasonlót kell találnunk:

```
# Az xdm futtatEsa 5- s futEszszinten
# Az xdm gy egy œj k l n szolgEltatEsz lesz
x:5:respawn:/etc/X11/prefdm -nodaemon
```

Jelentkezzünk ki és indítsuk újra az X-kiszolgálót, hogy az új beállítások életbe lépjenek. Ezután a munkaállomásról futtasdátok az `X -query ip_c m` (természetesen a kiszolgáló IP-címe) parancsot. Ha XDM-et futtatok, a dolognak működni kell. Természetesen sokan közületek a KDE-ben lévő `kdm` bejelentkezéskezőt használjátok, és csak egy üres szürke képernyőt láttok egy keresztel a közepén (bár a keresztet mindenfelé mozgatni lehet, ez nem túl érdekes). Ennek oka, hogy a KDM elérése az XDMCP-n keresztül egy további kis trükköt igényel.

A KDE a `kdmrc` fájlt használja. Az én rendszeremen ez a `/etc/kde/kdm` könyvtárban található. Keressétek meg az alábbi sorokat, és győződjétek meg róla, hogy az `[Xdmcp]` rész `Enable` kapcsolója `true`-ra van-e állítva:

```
[Xdmcp]
# Figyeljen-e a KDM az XDMCP kØrØsekre:
Enable=true
```

Mialatt François kihozza a vajas Brie sajtot a bor mellé, hadd szóljak néhány szót a legvékonyabb ügyfélgépekről, a merevlemez nélküli munkaállomásokról. Ezt a megközelítést különböző beállítások támogatják, ezek közül ma egyre fogok összpontosítani. Mivel ez a Linux munkahelyi bevezetésének egy igen fontos eszköze, ennek a kitűnő Linux-konyhának a tulajdonosaként kötelességemnek érzem, hogy más projektek közül

is kiemeljek egy maroknyit, amelyek figyelemre tarthatnak számot. Még záróra előtt megosztok veletek néhányat ezek közül, de most beszéljünk ezeknek a munkaállomásoknak az indításáról.

Az ethernetkártyátok megfelelő meghajtóprogramjával ellátott indítólemezt az Etherboot weboldalon (<http://etherboot.sourceforge.net>) található eszközök segítségével készíthettek. A forrás lefordítása egyszerűen a tar-csomagok kibontásából, az `src` könyvtárra való váltásból és a `make` parancs futtatásából áll. Egy még egyszerűbb módszer is rendelkezésünkre áll a ROM-o-matic.net (<http://rom-o-matic.net>) tagjainak köszönhetően. Ők foglalkoznak egy indító ROM-gyűjtemény karbantartásával, amelyet az Etherboot projekt alapján készítenek. Válasszuk ki a kívánt kiadást (a legfrissebb általában a legjobb), a kártyánk típusát (3c509, tulip stb.) és kattintsunk a **Get ROM** gombra. A letöltött fájl lemezre tudjuk írni:

```
cat eb-5.0.6-yournic.lzdisk > /dev/fd0
```

Voilà, a lemezünk kész! Tegyük be a munkaállomás számítógépbe, és indítsuk újra. A rendszer indítási folyamata tart egy kis szünetet, hogy lehetőséget adjon a helyi vagy hálózatos indítás kiválasztására – a hálózatos az alapértelmezett. A gond csak az, hogy az ügyfélgépnek indítás után nincs mihez csatlakoznia. Ez az a pont, ahol a Linux Terminal Server Project (<http://www.ltsp.org>) szerephez jut. A projekt lényege, hogy a vékony Linux-ügyfél beállítását és használatát a lehető legegyszerűbbé teszi.

Most tegyük félre egy kicsit a munkaállomást, és néhány percig összpontosítsunk a kiszolgálóra. Szükségünk lesz néhány további dologra, kezdve a DHCP-kiszolgálóval (Linux-rendszerünk `dhcp` csomagja), egy TFTP-kiszolgálóra (a legvalószínűbb, hogy `tftp-server` néven találjuk meg a telepítőlemezen), továbbá NFS-kiszolgálón csomagjainak a telepítésére. Mindegyiknek ott kell lennie a telepítő lemezen, úgyhogy nem kell túl messzire mennünk értük.

A TFTP esetén egy bejegyzés jön létre a `/etc/xinet.d` könyvtárban a szolgáltatás számára. Bár a telepítő létrehozta ezt a bejegyzést, alapértelmezésben sajnos le is tiltja. Győződjünk meg arról, hogy a szolgáltatás (`etc/xinet.d/tftp`) tartalmazza-e az alábbi sort:

```
disable = no
```

Indítsunk újra a `xinetd`-t a TFTP-démon működésbe hozásához. Ha még nem tettük meg, ez jó alkalom magunknak az LTSP-csomagok letöltésére. Az oldalon minden csomag tar és gzip formátumban is megtalálható, de előre fordított RPM-csomagokat is letölthetnek a Red Hat, SuSE, Mandrake stb. felhasználói. A Debian használók is megtalálják a maguk előre csomagolt DEB-fájljait. Ha a tar formátumot választjuk, egyszerűen bontsuk ki őket, és minden csomag esetén futtassuk le az `install.sh` parancsállományt. A csomagokról szólva, íme, ezekre lesz szükség a meghatározott sorrendben: `ltsp_core`, `ltsp_kernel`, `ltsp_x_core` és `ltsp_x_fonts`. Van még néhány helyi alkalmazáscsomag, ami szintén szükséges a futtatáshoz, például a Netscape, de ezekkel most nem foglalkozunk. Ez az a pont, ahol a beállítás során kezdtek egy kicsit érdekesebbé válni a dolgok. Három különböző gépre telepítettem fel az LTSP-t, és úgy tapasztaltam, hogy a sorrend igen fontos. Az LTSP beállításához már futó `dhcpd`-re volt szükségem, de a démon elindításához érvényes `/etc/dhcpd.conf` fájljal kellett

rendelkezem. Mivel az autoconfig nem működött, ez a fájl nem volt a gépen, ezért úgy döntöttem, hogy kézzel hozok létre egy saját *dhcpd.conf* fájlt – ez nem is olyan bonyolult. Vessetek egy pillantást az alábbi beállításra! Mintának megfelelő, természetesen mindig szem előtt tartva, hogy a tartomány (és a root-path, valamint a kiszolgáló IP-címe) különbözni fog az enyémtől:

```
subnet 192.168.22.0 netmask 255.255.255.0 {
    option domain-name "yourdomain.dom";
    option root-path
    ↪ "192.168.22.10:/opt/ltsp/i386";
    range dynamic-bootp 192.168.22.128
    ↪ 192.168.22.254;
    default-lease-time 21600;
    max-lease-time 43200;
}
host baroque {
    filename "/lts/vmlinuz-2.4.18-ltsp-1";
    hardware ethernet 00:50:FX:5B:XX:56;
    fixed-address 192.168.22.5;
    option host-name "baroque";
}
```

Vegyétek észre, hogy legalább egy számítógépet létrehoztam. A fájlnev megfelel a legfrissebb LTSP-kiadásnak, de az ethernetkártya eszköz- vagy MAC-címének (amellett, hogy itt csak kitalált érték szerepel) meg kell felelnie az ügyfélgép adatainak. A *dhcpd* a *service dhcpd start* parancsral indítható. Miután DEB-, RPM- vagy tar-csomagjainkat telepítettük, és elindítottuk a különböző szolgáltatásokat, ideje köszönetet mondanunk az LTSP tagjainak, mert a többi munka nagy részét már elvégezték helyettünk. Nekünk már csak a rendelkezésünkre álló előkészítő parancsfájlt kell lefuttatnunk:

```
cd /opt/ltsp/templates
./ltsp_initialize
```

A parancs beírása után figyelmeztetést kapunk, hogy a parancsfájl során fontos rendszerállományok fognak megváltozni, mégpedig a */etc/dhcpd.conf*, */etc/exports* és néhány egyéb fájl (lásd a *képet*).

Mivel az NFS új exportokkal frissült, és a DHCP beállításai is megváltoztak, mindkét demont újra kell indítani:

```
service nfs restart
service dhcpd restart
```

Csaknem készen is vagyunk, mes amis. Másodperceken belül rendelkezni fogunk egy élvonalbeli, mégis szerény, linuxos vékonyügyfél-telepítéssel. Mivel az LTSP minden programja a */opt/ltsp* könyvtárban van, néhány beállítás kedvéért egy kis kirándulást kell tennünk ide. Kezdjük a */opt/ltsp/i386/etc/lts.conf* fájl szerkesztésével. Nézzük az alábbi sort!

```
SERVER = 192.168.22.10
```

Ide annak a gépnek az IP-címét kell beírunk, amelyiket kiszolgálóként használjuk. Ezt az *lts.conf* fájlt érdemes közelebről is megismerni. Ebben lehetőség nyílik bizonyos beállítások megváltoztatására abban az esetben, ha a munkaállomások valamelyike nem működne az alapbeállításokkal. Íme egy beállítás, amit *baroque* nevű munkaállomás számára készítettem (egy

öreg soros egérrel kellett működésre bírnom):

```
XSERVER = vesa
X_MOUSE_PROTOCOL = "Microsoft"
X_MOUSE_DEVICE = "/dev/ttyS0"
X_MOUSE_BUTTONS = 3
LOCAL_APPS = N
SWAPFILE_SIZE = 64m
RUNLEVEL = 5
```

Amikor egyre újabb és újabb munkaállomásokat adunk a rendszerhez (és üzembe helyezzük), ne felejtjük el, hogy ezeket hozzá kell adnunk a */etc/dhcpd.conf* fájlhoz. Tehát a *dhcpd* szolgáltatást nem kell újraindítani a változtatások után. Amikor túlvagyunk ezen a kis kellemetlenségen, tegyük be a hajlékonylemezt a meghajtóba, indítsuk újra a munkaállomást, és néhány másodpercen belül látnunk kell a folyamatot, ahogy a rendszermag-lenyomatfájl a TFTP-n keresztül a munkaállomásra töltődik. Ezután gépindításkor lefutó sorokat láthatjuk, majd egy önműködő X-beállítás követi. Ha az X a folyamatnak ezen a pontján esetleg megáll, a munkaállomás bejegyzését az *lts.conf* fájlban átállíthatjuk 3-as futási szintre. Ekkor a gép indítása után szöveges módba jutunk.

A záróra vésszesen közeleg, mes amis, de megígértem, hogy megemlítek még néhány hasonló projektet. Figyelmetekre lehet érdemes a Solucorp's X terminal eszközkészlet (↪ <http://www.solucorp.qc.ca>), amelyet a Linuxconfról híres *Jacques Gélinas* vezet. Ne hagyjátok ki *Diego Torres Milano* weboldalát se a ↪ <http://pxes.sourceforge.net> címen, ahol a PXES vékonyügyfélcsoomagról szerezhetek további adatokat. Köszönöm, mes amis, hogy Chez Marcellel töltöttétek ezt az estét. Remélem, jó benyomást gyakorolt rátok a „vékony” Linux, és az üzleti életben hasznosítani tudjátok az itt látottakat. Az egyszerűsített felügyeletben rejlő költségmegtakarítás ezt a megoldást tökéletes választássá teszi a mai költségérzékeny világban. Frissítsük a kiszolgálót, és ezzel frissítettünk minden programot; készítsünk biztonsági másolatot a kiszolgálóról, és ezáltal mentettük az összes munkaállomást – mi lehetne ennél egyszerűbb? François, kérlek, utoljára még töltsd tele vendégeink poharát. Vizontlátásra a következő hónapig! A votre santé! Bon appétit!

*Linux Journal* 2002. november, 103. szám



*Marcel Gagné*

Mississaguában, Ontario államban él. Ő a szerzője a Kiskapu kiadásában szeptemberben megjelent Linux-rendszerfelügyelet (ISBN 963-9301-40) című könyvnek (jelenleg is egy könyvön dolgozik). A [mggagne@salmar.com](mailto:mggagne@salmar.com) címen érhető el.

### Kapcsolódó címek

Etherboot ↪ <http://etherboot.sourceforge.net>  
 Linux Terminal Server Project ↪ <http://www.ltsp.org>  
 Linux X Terminal Toolkit ↪ <http://www.solucorp.qc.ca>  
 Marcel's Wine Page  
 ↪ <http://www.marcelgagne.com/wine.html>  
 PXES Linux Thin Client ↪ <http://pxes.sourceforge.net>  
 ROM-o-matic ↪ <http://rom-o-matic.net>