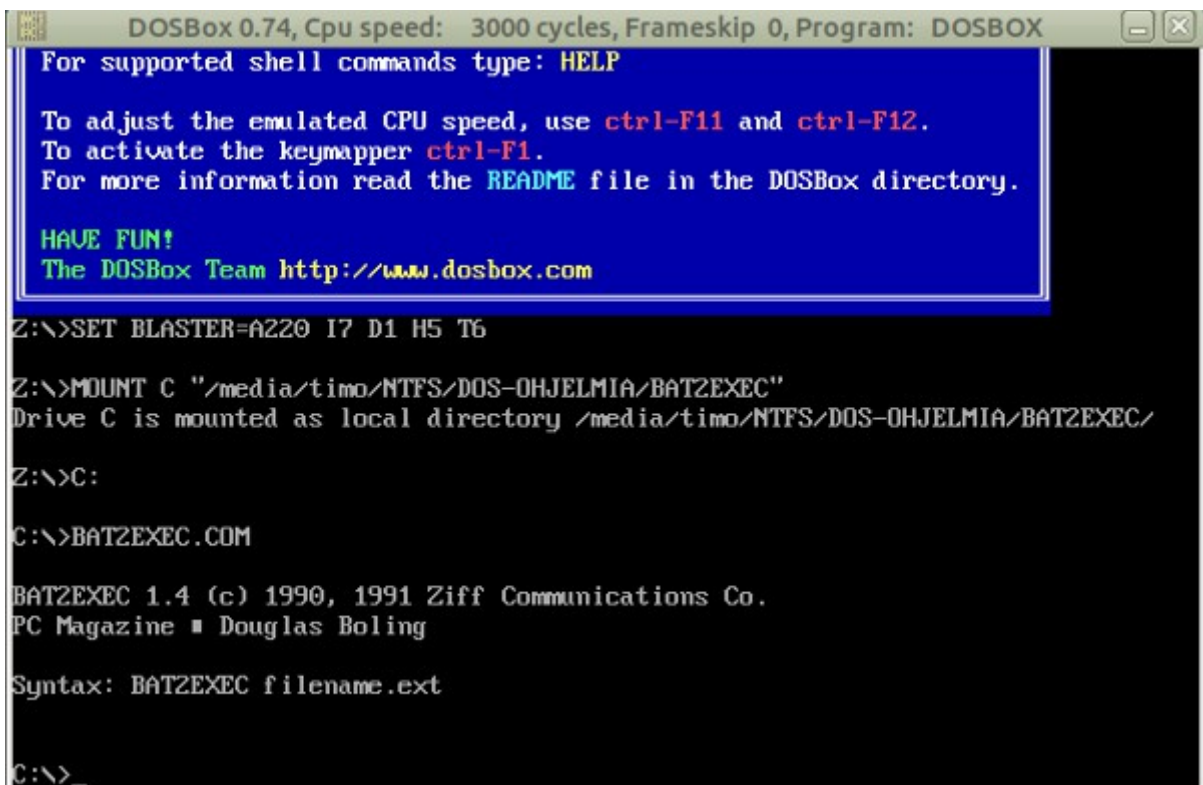


Timo Kinnunen
Särkiniementie 16 A 41
70700 Kuopio
Finland

UNP 3.15, Bat2Exec, ja ST2DOS -apuohjelmien, ja joidenkin tiivistysohjelmien opas

Koska joissakin piireissä on ilmennyt kiinnostusta vanhoihin käyttöjärjestelmiin, ja vanhoihin sovelluksiin, kerron tässä oppaassa eräistä välineistä, joita takavuosina käytettiin runsaasti. Näistä on apua esimerkiksi niille, jotka haluavat luoda eräajoistaan omia itsenäisiä ohjelmia, tai lukea *Atari ST* -tietokoneella alustettuja levykkeitä DOS -järjestelmässä, tai "pakata" sovelluksia, tai "purkaa" niitä takaisin alkuperäiseen kokoonsa. Tiedot ovat aika teknisiä, mutta se johtuu itse asian luonteesta. En ole varma siitä, saako näitä sovelluksia enää mistään, mutta se on ainakin varmaa, että ellei niitä osaa lainkaan kysyä, ei niitä voi myöskään löytää. Ja onhan tämä tarkoitettu asian harrastajille, mutta ei veljille svengissä. Ja tarkista ihmeessä tämän oppaan lopussa oleva "**kätilläk**" -laulu, sillä jos se nyt jotakuta naurattaa, niin voi sanoa, että hovit on tosi halvat. Mutta jos naurattikin, niin kyllähän se minuakin on huvittanut.

Bat2Exec ohjelma ja eräajojen muuntaminen ajettaviksi ohjelmiksi



```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX
For supported shell commands type: HELP
To adjust the emulated CPU speed, use ctrl-F11 and ctrl-F12.
To activate the keymapper ctrl-F1.
For more information read the README file in the DOSBox directory.
HAVE FUN!
The DOSBox Team http://www.dosbox.com
Z:\>SET BLASTER=A220 I7 D1 H5 T6
Z:\>MOUNT C "/media/timo/NTFS/DOS-OHJELMIA/BATZEXEC"
Drive C is mounted as local directory /media/timo/NTFS/DOS-OHJELMIA/BATZEXEC/
Z:\>C:
C:\>BATZEXEC.COM
BATZEXEC 1.4 (c) 1990, 1991 Ziff Communications Co.
PC Magazine ■ Douglas Boling
Syntax: BATZEXEC filename.ext
C:\>_
```

DOS -järjestelmässä käytetyt **BAT** -eräajot ovat tekstimuotoisia, ja sellaisina niiden ajaminen voi olla hidaskäyttöprosessi, koska DOS suorittaa jokaisen rivin kohdalla erikseen kääntämisen konekielelle. Monet etsivätkin aikoinaan keinoja, joilla eräajojen suorittamia prosesseja voitaisiin nopeuttaa. Jos olet itse etsinyt välinettä, jolla voisit muuntaa **BAT** -eräajot ajettaviksi ohjelmiksi, saattaa vieläkin jostakin löytyä **Bat2Exec** -kääntäjäsovellus, jolla tällainen muunnos voidaan suorittaa helposti. Käytännössä tämä merkitsee sitä, että voit tehdä "ohjelman" helposti tekstimuodossa, kunhan vain opettelet DOS -järjestelmän internaaliset käskyt, ja validit komentojen muodot. Ennen varsinaista käännettä voit

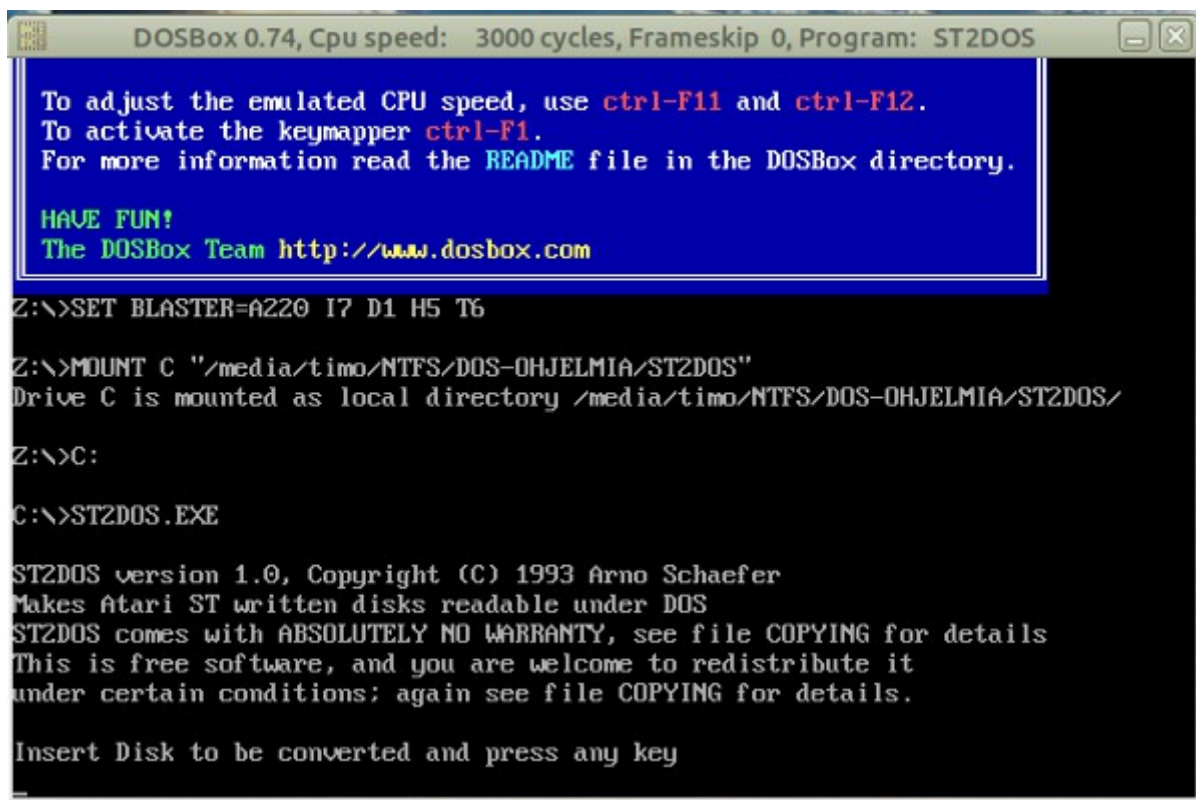
kokeilla "ohjelmaasi" vielä sen ollessa tekstimuodossa, ja jos olet tulokseen tyytyväinen, voit muuttaa sen **COM** -ohjelmaksi, jolloin siitä tulee paljon nopeampi.

Bat2Exec ohjelman käyttötarkoitus

Doug Bolingin vuonna 1990 julkaisemalla **Bat2Exec** (v9) -kääntäjällä voit muuntaa eräajotiedostosi (**BAT**) nopeammin toimiviksi **COM** -ohjelmiksi. Varsinkin suuret eräajotiedostot ovat usein käytännössä tavattoman hitaita. Tällainen käsittely on perusteltu jos olet luonut suurella vaivannäöllä pitkiä komentoketjuja, joita voit käyttää samanlaisina myös pitkälle tulevaisuuteen, ja joilla hallitaan mutkikkaita prosesseja, ja niitä varsinaisesti korvaavia menetelmiä ei ole olemassakaan, tai ne ovat kalliita. Kääntäjän käyttö sinällään on tavattoman helppoa. Kirjoita vain komentoriviltä oletushakemistossa, jossa sekä kääntäjä -että käsiteltävä eräajo sijaitsevat: *Bat2Exec tiedosto.BAT* - jossa "*tiedosto.bat*" on muunnettavan eräajon nimi. **Bat2Exec** -kääntäjä luo siitä muuntoprosessissa ohjelmana ajettavan **COM** tiedoston ("*tiedosto.com*"). Periaatteessa eräajon muodollinen rakenne säilyy ennallaan pienin muutoksin, mutta tiedostoon lisätään suorittava osa, jonka ansiosta DOS voi ajaa sen ohjelman tapaan. Tulostiedosto on siten kooltaan suurempi kuin alkuperäinen tiedosto. Mikäli **Bat2Exec** ei löydä muunnettavaa tiedostoa antamallasi nimellä, tulee näytölle siitä virheilmoitus, ja jos **Bat2Exec** ei taas ymmärrä jotakin eräajon riviä, ilmaantuu näytölle viesti, jossa viitataan kyseiseen virheelliseen riviin. Tämä kaikki viittaa siihen, että kyseessä on varsin automaattinen ohjelmasovellus. Tällä kääntäjäohjelmalla ei kuitenkaan saa koskaan käsitellä **AUTOEXEC.BAT** -tiedostoa, koska **COMMAND.COM** komentotulkki olettaa sen olevan tavallisen tekstitiedoston, ja komentotulkki nimenomaan "kääntää" itse sen symboliset komennot konekielille tietokoneen käynnistyksen yhteydessä. Sama pätee myös moniin DOS -järjestelmälle aikanaan tehtyihin sovelluksiin, joiden parametrit asetettiin **BAT** eräajojen avulla. Tällaisia olivat mm. **Xerox Ventura 2.0** ja **PBrush 1.2**. Samoin ei ohjelmalla tulisi käsitellä sellaisia eräajoja, joissa käynnistetään esimerkiksi muistiresidenttejä (*TRS*) ohjelmia. Syynä tähän on DOS järjestelmän muistinhallinnan rakenne: jos *TSR* ajetaan siitä ohjelmasta, joka on luotu **Bat2Exec** -ohjelmalla, ei tämän ohjelman itsensä käytössä ollut muisti vapaudu kun ohjelman käyttö lopetetaan, tai se saavuttaa päätepisteensä. **Bat2Exec** -kääntäjällä luotu ohjelma käyttäytyy hieman eri tavallakuin alkuperäinen eräajo käyttäytyisi. **COM** tiedosto ei heijasta jokaista riviä näytölle eräajon tapaan, ja näytölle ei ilmaannu myöskään: "*Terminate Batch file (y/n)*" - viestiä vaikka *CRTL-BREAK* näppäinyhdistelmää olisi painettu **BREAK** -tilan ollessa päällä. Ohjelma yksinkertaisesti vain päättää tässä tapauksessa toimintansa mitään sen kummemmin ilmoittelematta. **Bat2Exec** -kääntäjää ei kannata käyttää jos käytössä on nopea kone, tai sellainen käyttöjärjestelmä, jonka toiminta ei perustu eräajoihin, tai jos järjestelmän itsensä muistinhallinta on sellaista, ettei siihen voi vaikuttaa yksittäisillä sovelluksilla. Siten ei ole järkevää käyttää **BAT** eräajoja ainakaan **Windows 95** käyttöjärjestelmässä, vaikka ne siinä toimivatkin. Myöskään **OS/2 Warp 3.0** järjestelmässä ei näiden käyttö kannata, ja siinä kannattaakin sen asemesta opetella *REXX* -kieli, ja luoda sillä komentotiedostoja (jos tätä järjestelmää nyt enä kukaan käyttää). Jos käyttöjärjestelmänä on DOS, ja tiedostojärjestelmänä vanha *FAT*, ja käytät runsaasti erilaisia eräajoja, ne syövät todella tehokkaasti levyresursseja, koska useimmat eräajotiedostot ovat pieniä, mutta varaavat aina silti järjestelmän mukaisen pienimmän varausyksikön jokaista luotua pientä tiedostoa varten. Levykoon kasvaessa suurenee tämä tila vastaavasti, ja tilaa tuhlautuu enemmän. Tämä ongelma ei korjaannu **Bat2Exec** -kääntäjällä, mutta sen etuna on se, että vahingossa tapahtuva **BAT** -eräajojen laajamittainen postu, tai "päivitykset" eivät turmele prosessia. Käytännössä saatu etu ohjelmamuodon kohdalla on se, että se on nopeampi kuin tekstimuotoiset **BAT** eräajot, vaikka se ei tilaa säästäkään. Sillä, että kääntäjällä luodut **COM** -ohjelmat ovat tiedostokooltaan suurempia kuin alkuperäinen **BAT** -tiedosto, ei ole merkitystä, koska ne kuitenkin tavallisesti ovat pienempiä kuin pienimmän varausyksikön koko, tai ne eivät kuitenkaan vie moniakaan varausyksikköjä. Jos esimerkiksi muutetaan eräajo, jossa on vaikka yksinkertainen *REM* -lause, on tuloksena 68 -tavun suuruinen **COM** tiedosto, joka sisältää *setup*- ja *lopetusproseduurit*. Ohjelman koko lisääntyy kuitenkin nopeasti lisättäessä rutiinien määrää, ja tällaisen tiedoston kokoa ei voida kasvattaa yli 64 kilotavun, koska se on **COM** tiedoston suurin mahdollinen koko. Tästä rajoituksesta johtuen **Bat2Exec** -kääntäjällä luodut **COM** tiedostot eivät tarvitse - ja eivät sisällä erillistä "*relocation*" -taulua, jollaiset ovat välttämättömiä suurissa **EXE** tiedostoissa, ja liittyvät niiden erilaiseen muistinhallintaan. Luotu ohjelma voidaan tosin tiivistää esimerkiksi **TINYPROG** -ohjelmalla, joka muuttaa se tavalliseksi samalla **EXE** tiedostoksi, ja tekee siinä käytetyt komennot sellaisiksi, ettei niitä voida enää lukea levyeditoreilla. Ohjelma toimii tämän muunnoksen jälkeenkin

normaalisti. **Bat2Exec** -kääntäjällä tehtyjen **COM** ohjelmien puute on se, että ne eivät osaa mukautua järjestelmässä tehtyihin muutoksiin, kuten uuteen hakemistolliseen järjestelyyn - ja siksi ne saattavat toimia väärin, ja jopa tuhoisalla tavalla myöhemmin. Tästä syystä niihin on syytä lisätä erilaisia tarkistuksia koskien mm. tiettyjä avaintiedostoja, joiden tulisi löytyä tietyistä hakemistoista. Huomaa, että on olemassa sellaisiakin ohjelmia, joilla voidaan lisätä käytettävissä olevien eräajo-funttioiden määrää - kuten **Michael Meffordin** *BATCHMAN*.

ST2DOS (Versio 1.0)



```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: ST2DOS

To adjust the emulated CPU speed, use ctrl-F11 and ctrl-F12.
To activate the keymapper ctrl-F1.
For more information read the README file in the DOSBox directory.

HAVE FUN!
The DOSBox Team http://www.dosbox.com

Z:\>SET BLASTER=A220 I7 D1 H5 T6

Z:\>MOUNT C "/media/timo/NTFS/DOS-OHJELMIA/ST2DOS"
Drive C is mounted as local directory /media/timo/NTFS/DOS-OHJELMIA/ST2DOS/

Z:\>C:

C:\>ST2DOS.EXE

ST2DOS version 1.0, Copyright (C) 1993 Arno Schaefer
Makes Atari ST written disks readable under DOS
ST2DOS comes with ABSOLUTELY NO WARRANTY, see file COPYING for details
This is free software, and you are welcome to redistribute it
under certain conditions; again see file COPYING for details.

Insert Disk to be converted and press any key
```

ST2DOS -ohjelman on vuonna 1993 tehnyt **Arno Schaefer**, ja sen tarkoituksena on muuntaa *Atari ST* tietokoneella luotuja levykkeitä niin, että niitä voidaan lukea DOS -järjestelmässä. Nämä tiedot auttavat niitä, joilla on tällä tietokoneella kirjoitettuja dokumentteja levykkeillä, mutta käytettävissä ei ole enää *Ataria*, vaan jokin PC, jossa on jokin DOS -järjestelmä. **ST2DOS** ohjelmalla voidaan siis tehdä *Atari ST* -järjestelmällä alustetuista levykkeistä luettavia myös DOS järjestelmässä. **ST2DOS** on *GNU* ohjelma, eli sitä voidaan jakaa vapaasti, sekä myös muokata niiden rajoitusten puitteissa, joita **Free Software Foundationin** "*GNU General Public License*" on määrittänyt. Sellaisen pitäisi kuulua alkuperäiseen jakeluarkistoon. Mutta vapaudesta on se tavanomainen hintansakin: ohjelman aiheuttamista vahingoistakaan tekijä ei vastaa.

Miksei TOS alustettujen levykeiden luku onnistu DOS -järjestelmässä?

Levykkeet, joita on alustettu *Atari ST* tietokoneella, jossa on **TOS** -käyttöjärjestelmä, joka on vanhempi kuin 1.4, eivät ole suoraan luettavissa DOS -järjestelmässä. Tämä johtuu näiden järjestelmien erilaisesta tavasta käsitellä levykkeitä.

1. **TOS** ei kirjoita DOS -järjestelmän kannalta oikein alkulatauslohkossa sen *Media Descriptor* -bittiiä (**Offset 15h**), eikä ensimmäin bitti kahdessa levykkeellä olevassa tilanvaraustaulussa (*FAT*) ole sekään kirjoitettu oikein. Sen pitäisi olla **F8H** yksipuoleisissa levykkeissä, ja **F9H** kaksipuoleisissa.
2. Hyppykäsky alkulatauslohkon ensimmäisessä kolmessa bitissä on erilainen **TOS** ja DOS -järjestelmissä, mikä johtuu *Atari ST* ja PC -koneissa olevien prosessorien erilaisuudesta. DOS

-järjestelmää varten hyppyykäslyn pitäisi olla **EB 3C 90**. Tämä "ristiriita" ei kuitenkaan ole mikään ongelma **TOS** järjestelmälle.

ST2DOS ohjelma kirjoittaa *Atari ST* -koneilla alustetulle levykkeelle oikeat arvot, ja siitä syystä levykkeitä voidaan lukea **DOS** järjestelmässä. Kuitenkaan ohjelmaa ei ole tarkoitettu *High Density* -levykkeiden käsittelyyn, mikä johtuu siitä, että **TOS** versio 1.4 aikana ei tällaisia levykkeitä vielä ollut käytettävissä.

ST2DOS ohjelman käytöstä ja ongelmista

Ohjelman käytössä on omat niksinsä, ja sitä käytetään **DOS** -järjestelmässä, jossa se kysyy, mitä levykeasemaa käytetään, jos niitä on kaksi. Ohjelma pitää pysäyttää *CTRL-C* -näppäinyhdistelmää painamalla, koska se suorittaa kerran käynnistetyksi tultuaan muutoin loputonta silmukkaa. Ongelmiakin ohjelman käytössä on, sillä sitä ei ole testattu kaikissa erilaisissa levykeformaateissa, ja se saattaa epäonnistua levykkeillä, joissa on enemmän kuin yhdeksän sektoria, mutta tämä riippuu koneen *BIOSISTA*, sekä siitä, onko levyke yksipuoleinen. Yleisesti kannattanee käyttää vain vanhempia **PC** koneita näissä operaatioissa. Huomaa, että jos käynnistät tietokoneesi uudelleen operaation jälkeen, ja levyke on edelleen levykeasemassa, ei käynnistyminen onnistu, koska tarvittavaa alkulatausohjelmaa ei tietenkään löydy, vaikka levyke onkin "lukukelpoinen". Jos kyseessä on levyke, joka on bootattavissa *Atari ST* koneessa, ei se enää onnistu tällaisen operaation jälkeen.

Virheviestit

'Invalid Bootsector'

Tämä viittaa siihen, ettei alkulatauslohkon sivujen määrä ei ole 1 tai 2.

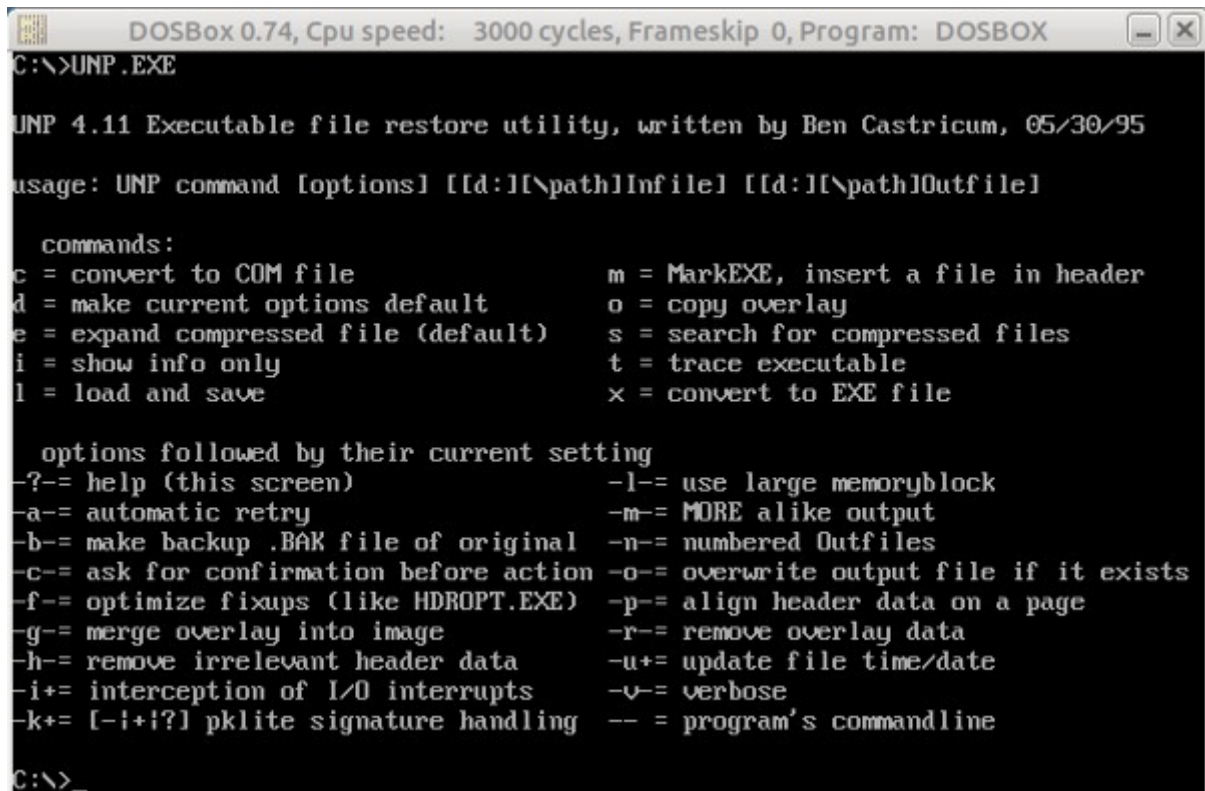
'Error reading Bootsector/FAT 1/FAT 2'

ST2DOS ei voinut löytää alkulataussektoria, mikä johtuu siitä, ettei levykettä ole levykeasemassa, tai siitä, että sektori on viallinen.

'Error writing Bootsector/FAT 1/FAT 2'

Tämä tarkoittaa sitä, ettei sektoriin ole voitu kirjoittaa mitään - mikä saattaa johtua siitä, että levykkeen kirjoitussuojaus on päällä.

UNP 3.15 ohjelma ja sen käyttötarkoitus



```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX
C:\>UNP.EXE

UNP 4.11 Executable file restore utility, written by Ben Castricum, 05/30/95

usage: UNP command [options] [[d:][\path]infile] [[d:][\path]outfile]

  commands:
c = convert to COM file           m = MarkEXE, insert a file in header
d = make current options default  o = copy overlay
e = expand compressed file (default) s = search for compressed files
i = show info only               t = trace executable
l = load and save                x = convert to EXE file

  options followed by their current setting
-?-= help (this screen)          -l-= use large memoryblock
-a= automatic retry              -m= MORE alike output
-b= make backup .BAK file of original -n= numbered Outfiles
-c= ask for confirmation before action -o= overwrite output file if it exists
-f= optimize fixups (like HDROPT.EXE) -p= align header data on a page
-g= merge overlay into image      -r= remove overlay data
-h= remove irrelevant header data  -u+= update file time/date
-i+= interception of I/O interrupts -v= verbose
-k+= [-i+!?] pklite signature handling -- = program's commandline

C:\>_
```

Ben Castricumin tekemä **UNP V3.15** ohjelma (ylläolevassa kuvassa UNP 4.11) on tarkoitettu sekä DOS -järjestelmässä käytettäväksi, mutta se toimii myös **Windows 95** komentokehoite -tilassa. Ohjelman käyttötarkoitus on laajentaa tiivistettyjä ohjelmatiedostoja takaisin niiden tiivistämättömään muotoon. Useat ohjelmatiedostot on käsitelty niin, että ne ovat tiivistettyjä jollakin pakkerilla. Erityisen sovelias **UNP V3.15** ohjelma on vanhojen DOS-pohjaisten esitysohjelmien käsittelyyn, joiden kokoa on pienennetty levytilan säästämiseksi. Tehokkainta levytilan säästö oli käsiteltäessä tiivistysohjelmilla sähköisiä julkaisuja - eli takavuosina paljonkin käytettyjä "elektronisia sanomalehtiä ja kirjoja". Jos nämä haluttaisiin jostakin syystä nyt sellaiseen muotoon, jota voisi käsitellä edelleen esimerkiksi tekstinkäsittelyohjelmalla, tai saattaa esimerkiksi HTML dokumentin muotoon - pitäisi esitysohjelma laajentaa ensin takaisin alkuperäiseen kokoonsa, ja senjälkeen "poimia" sitä teksti vaikkapa **Vernonin List** -ohjelmalla. **UNP V3.15** -ohjelma yrittää suorittaa periaatteessa täysin vastakkaisen toimenpiteen minkä esimerkiksi ohjelmat **PKlite.EXE** tai **Lzexe.EXE** ovat tiivistetylle ohjelmalle tehneet. Nämä kyseiset tiivistysohjelmat soveltavat datan tiivistystä "ajettavissa oleviin" (*Executable*) ohjelmatiedostoihin niin, että ne pakkautuvat fyysisesti pienempään muotoon, ja sisältävät yleensä koodin, jonka avulla ne laajentuvat keskusmuistissa alkuperäiseen kokoonsa. Tiivistysohjelmat jättävät siis käsittelemänsä ohjelmatiedostot sellaiseen tilaan, että niitä voidaan ajaa entiseen tapaan. Latauksen, ja ajon välillä suoritetaan tiedostonpurku tietyn algoritmin mukaan, ja koska tiedostoon ei yleensä tallenneta tässä prosessissa mitään, ei vastakkaisuuntaista operaatiota tarvita. Kuitenkin: jos puhutaan lataukseen kuluvan ajan piteneemisestä tai lyhenemisestä tiivistettäessä tiedostoja, tai palautettaessa niitä entiseen kokoonsa, on kyse vain muutamien sekunnin sadasosien aikaerosta. Jos käytetään nopeaa tietokonetta, on kai lähes sama ovatko tiedostot tiivistettyjä vai eivät. Ainoaksi eduksi jää sittenkin säästö levytilassa ... ja senkin tekijän merkitys vähenee sitä mukaa kun levytila halpenee. **UNPV3.15** kykenee muunkaltaisia rutiineita tiivistetyistä ohjelmista. Esimerkiksi **Central Point Anti-Virus (tm)** -immunisaatiokoodeja voidaan sillä turvallisesti poistaa.

UNP 3.15 ohjelman ominaisuuksia ja vaatimuksia

UNPV3.15 ohjelman käyttö vaatii vähintään [vain] 80086 -prossessorin. Jos halutaan saada ohjelmasta täysi hyöty, on vähintään **MS-DOS 5.0** tai tätä uudempi DOS -käyttöjärjestelmä suositeltava, koska **UNP**

V3.15 kykenee käyttämään 640 kilotavun perusmuistin yläpuolella olevaa ylämuistialuetta (*Upper Memory Block* =UMB). Toisaalta ohjelma ei itse vaadi kovinkaan paljoa muistia - vain noin 15 kilotavua. Tähän on kuitenkin laskettava lisäksi kohdeohjelmien käsittelyyn tarvittava muistitila. Mutta ennekuin aloitat **UNP V3.15** ohjelman käytön, täytyy sinun huomioida muutamia seikkoja. Jos ohjelmatiedostoa on tiivistetty, eivät sen *ohjelmakäynnistyksen jälkeen haettavat osat*, eli ohjelman "*overlays*" -osat ehkä toimi oikein senjälkeen kun ohjelma on laajennettu takaisin entiseen kokoonsa. Tämä johtuu siitä, että ohjelman laajentaminen alkuperäiseen kokoonsa laajentaa myös **EXE** -tiedoston *koodikokoa*, mistä seuraa se, että mainitut "*overlay*" elementit siirtyvät muistissa ylöspäin. Jotkut ohjelmat eivät tarkista sitä, missä kohden jokin tietty "*overlay*" on sillä hetkellä kun sitä tarvitaan, ja jos ohjelmalla ei ole tätä tarkistusta, ei voi myöskään tapahtua mitään. Tästä syystä: aina kun käytät **UNP V3.15** ohjelmaa konvertoidaksesi tiedostoa toisenlaiseksi rakenteeksi, ota huomioon, etteivät ohjelmat enää toimi täsmälleen samanlaisissa olosuhteissa kuin aikaisemmin. Vaikka nämä mainitut erot eivät aiheutakaan todennäköisesti mitään ongelmia useimpien ohjelmien kohdalla, on silti olemassa aina myös ohjelmia, jotka edellyttävät olosuhteiden pysymistä vakioisina toimiakseen. Yksinkertaisin keino suojautua siltä, että menetettäisiin jokin ohjelma käsittelyn seurauksena on ottaa siitä varmuuskopio, tai käyttää **UNP V3.15** ohjelmaa '-b' -kytkimellä - jolloin **UNP V3.15** tekee käsiteltävästä ohjelmasta varmuuskopion, jonka se nimeää tarkenteella **BAK**. Tavallisesti - jos ongelmia ilmenee, voidaan ohjelma palauttaa kirjoittamalla sille uudelleennimeämisen kautta alkuperäinen **EXE** -tarkenne - kun sitä ennen on tuhottu ongelmia aiheuttava, laajennettu **EXE** ohjelmatiedosto.

UNP V3.15 ohjelman käyttö

Ohjelmaa **UNP V3.15** käytetään DOS komentoriviltä. Kun halutaan tietoa ohjelman käytöstä annetaan komentoriviltä komento:

UNP

tai

UNP ?

Komentoriville kirjoitettavaa komentoa voidaan sketsinomaisesti kuvata seuraavasti:

Käyttö: *UNP komento [optiot] [d:][/polku]Syöttötiedosto [[d:][/polku]Tulostustiedosto]*

Komennot

e = laajentaa tiivistetyn tiedoston (oletus)

Tämä komento laajentaa tiivistetyn tiedoston. Jos et spesifioi komentoa, käyttää **UNP** tätä oletuksena. Jos komentoa käytetään ilman tiedostomääritystä, **UNP** konvertoi kaikki oletushakemiston tiedostot.

c = konvertoi COM tiedostoksi

Jotkut **EXE** tiedostot voidaan konvertoida **COM** tiedostoiksi. Tämä voidaan tehdä käyttämällä tätä komentoa. Ennenkuin tätä komentoa käytetään, tulisi olla tarkkaan selvillä siitä, mitä tekee - eli voidaanko tiedosto muuntaa tällä tavoin, ja onko tuloksena toimiva ohjelma. **MS-DOS** ei tunnista tiedostoja niiden tarkenteen mukaan **COM** tai **EXE** tiedostoiksi, vaan järjestelmä etsii esimerkiksi **EXE** tiedostosta headeriä '*MZ*', tai '*ZM*', sekä sitä seuraavia yksityiskohtaisempia tietoja muista ominaisuuksista. Tiedämme myös, että **COM** tiedoston koko on rajattu, eli sen on oltava alle 64 kilotavua, mikä on sama kuin Windowsin **NotePad.EXE** ohjelman suurin mahdollinen käsiteltävä tiedosto, tai Windowsin **INI** -tiedostojen suurin mahdollinen koko. Näistä, ja eräistä muista seikoista voimme "tunnistaa" aika pitkälle mikä on tiedoston "oikea" tarkenne.

i = vain info

Käytä tätä komentoa jos haluat vain joitakin tietoja tiedostosta. **UNP** näyttää kaiken informaation kuten käytettäessä '**E**' -komentoa, mutta ei laajenna tiedostoa, tai tee siitä varmuuskopiota.

l = lataa muistiin ja tallettaa, ei tiivistystä (vain EXE tiedostot)

Tämä komento lataa **EXE** tiedoston muistiin, mutta ei laajenna sitä alkuperäiseen kokoonsa - vaan se pelkästään kirjoitetaan levyille. Komento on käyttökelpoinen jos haluat poistaa "**overlay**" -osan, tai irrelevantin header-datan. Käytä tätäkin komentoa varoen, koska aina ei ole tarkoituksenmukaista poistaa "**overlay**" -osaa.

s = etsii tiivistettyjä tiedostoja

Kun käytät tätä komentoa, luodaan vain pieni luettelo niistä tiivistetyistä tiedostoista, jotka vastaavat antamaasi tiedostomääritystä.

x = konvertoi EXE tiedostoksi

Jotkut tiivistysohjelmat voivat tiivistää vain **EXE** tiedostoja (kuten **Lzexe.EXE**). Tätä komentoa käyttäen voit konvertoida **UNP V3.15** ohjelmalla **COM** tiedostoja **EXE** tiedostoiksi. Tiedosto kirjoitetaan **UNP** ohjelmassa levyille oletusarvoisesti aina **EXE** -muodossa.

Optiot

-? = apua (tämä ruutu)

Saadaksesi luettelon **UNP** komennoista, tai optioista, käytä tätä kytkintä. Tällöin muut mahdollisesti samalla esitetyt komennot tai kytkimet jätetään huomiotta.

-a = automaattinen uudelleenyritys

Joitakin tiedostoja on muutettu mahdollisesti useammin kuin kerran. Tämä kytkin pakottaa **UNP** ohjelman käsittelemään tiedostoa uudelleen sen jälkeen kun sitä on kerran muutettu. Kytkin on hyödyllinen myös jos haluat laajentaa tiedoston joka on *Central Point AntiViruksen* saastuttama. Huomaa, että monet virusohjelmista lisäävät oman koodinsa tiedostoon, jolloin sen koko samalla hieman kasvaa. On tavattoman vaikeaa muuttaa tiedostoa niin että sen alkuperäinen koko ei edes jonkin verran muuttuisi. Mutta tiedosto voi olla täysin vaaraton, vaikka sen koko olisikin hieman kasvanut. Myös **UNP V3.15** lisää tiettyjä tiedostoja käsitellessään siihen '**PK**' signatuurin, joka ei ole aito, koska **PKlite.EXE** lisää tiedostoon tällaisen koodin, josta ohjelma "tietää" olevansa **PKlite.EXE** ohjelman tiivistämä. Ohjelma saattaa "vaatia" tämän '**PK**' -signatuurin senkin jälkeen kun se on laajennettu takaisin alkuperäiseen kokoonsa.

-b = tee alkuperäisestä varmuuskopio tarkenteella .BAK

Jos haluat säilyttää alkuperäisestä tiedostosta varmuuskopion, käytä tätä kytkintä. Alkuperäinen tiedosto nimetään uudelleen tarkenteella **BAK**.

-c = kysely vahvistuksesta ennen laajennusta

Tämä pakottaa **UNP** ohjelman kysymään sinulta vahvistusta sille laajennetaanko tiedosto vai ei - aina kun **UNP** löytää uuden tiivistetyn tiedoston. Tämä on erittäin kätevä piirre jos käsitellään samalla kertaa useita ohjelmatiedostoja.

-h = poista irrelevantti headerin data

Useat linkkiohjelmat lisäävät **EXE** headeriin hyödyttöä dataa. Tämä kytkin poistaa kaiken tällaisen hyödyttömän informaation ja pienentää samalla headerin kokoa.

-i = älä keskeytä toimintaa vaikka INT 21h -kutsu esitetään

Oletusarvona **UNP** tarkkailee DOS -keskeytyksiä (21h) tarkistaakseen toimiiko ohjelma odotetulla tavalla. Niinpä mikä tahansa odottamaton *INT 21h* -kutsu saa **UNP** ohjelman keskeyttämään toimintansa. Jos sinulla on käytössä poikkeavia *TRS* -ohjelmia, käytä tätä kytkintä. **UNP** ohjelman tekijä on itse käyttänyt sitä käyttäessään *Turbo Debuggeria*.

-k = pklite signatuuri; - = älä lisää, + = lisää aina, ? = kysy

Tämän kytkimen avulla voit käsitellä pklite '*PK*' -signatuuria. Kyseinen signatuuri lisätään yleensä tiedoston alkuun, ja sellaisen lisää aina **PKlite.EXE** ohjelma käsittelemiinsä tiedostoihin. **UNP** ohjelmaa käytettäessä tarjolla on kolme eri vaihtehtoa:

-k- = pklite signatuuria ei lisätä. Näin käy myös jos käytät pelkästään '**-k**' -kytkintä.

-k+ = lisää aina pklite signatuuri. Tämä vaihtoehto on **UNP** ohjelmassa oletusarvona, joten voit käyttää pelkästään '**-k**' -kytkintä jos haluat **UNP** ohjelman lisäävän signatuurin.

-k? = kun käytät tätä vaihtoehtoa **UNP** kysyy toimenpiteitä aina kun se löytää signatuurin.

-l = käytä aina kiinteää latausta (loadfix)

Aina **UNP** versiosta 3.12 lähtien ohjelma ei ole täyttänyt perusmuistista sen ensimmäistä 64 kilotavun osaa (koska näin voidaan käsitellä suurempia tiedostoja). Kun **UNP** löytää tiedoston, joka vaatii kiinteää latausta, se lataa sen uudelleen tähän 64 kilotavun osaan. Jos suunnittelet laajentaa useita **EXEPACK.EXE** ohjelmalla tiivistettyjä tiedostoja, saatat haluta, ettei tätä uudelleenlatautumista tapahdu. Itse asiassa tällä ei ole muuta vaikutusta kuin että se hieman nopeuttaa **UNP** ohjelman toimintaa.

-o = päällekirjoita tulostustiedosto, jos sellainen on olemassa

Jos haluat kohdetiedoston päällekirjoitettavaksi, voit ohittaa tätä koskevan kyselyn tällä kytkimellä. Käytä tätä kytkintä van jos olet täysin varma mitä teet, sillä saatat näin menettää alkuperäisen tiedoston.

-p = align header data on a page

On sanottu, että jos **EXE** tiedoston headeri on kaksi kertaa niin suuri kuin 512 bittiä, se latautuu nopeammin. (Tässä voi olla järkeäkin, koska yksi sektori on myös 512 bittiä). Tämä kytkin tekee mainitun kertoman.

-r = poista "lisätyn osan" (overlay) data

Jos **EXE** tiedostossa on osioita, jotka ohjelma hakee tarvittaessa myöhemmin käyttöönsä, mutta ei lataa niitä muistiin kun ohjelma muutoin ladataan, sanotaan näitä osia "*overlay*" -elementeiksi. Tätä kytkintä käyttämällä voit sallia tulostustiedoston koon olevan saman kuin ladatun kuvan. Tällä menetelmällä pudotetaan pois kaikki, mikä jää ulkopuolelle. Huomaa kuitenkin, että "*overlay*" -osat voivat sisältää kaikenlaista dataa, joten tämän osan poistamisen myötä saattaa poistua myös jotakin hyödyllistä, kuten se, että jotkut tiivistysohjelmat tallentavat tällaiseen "osaan" itse ohjelman tiivistetyn version.

-u = päivitä tiedoston aika/päivämäärä nykyhetkeen

Oletusarvona **UNP** asettaa tiedoston ajan ja päivämäärän samaksi kuin mitä se oli lähdetiedostossa. Jos kuitenkin haluat päivitystä nykyhetkeen, käytä tätä kytkintä. Kysymys ajasta ja päivämäärästä voi kuitenkin olla pulmallinen, jos puhutaan jostakin muusta kuin DOS-ympäristöstä, joka käyttää aina

paikallista aikaa - mikä käytännössä tarkoittaa sitä, että jokin tietty aikavyöhyke on käytössä. Tällainen järjestelmä on myös **Windows 95** järjestelmässä, vaikka siinä voidaankin vaihtaa aikavyöhykettä, mutta kerrallaan käytössä voi niistä olla vain yksi. Ongelmaa ei ole, jos tiedostoja ei siirretä vyöhykkeeltä toiselle sellaisenaan, ja samalla järjestelmästä toiseen. Jos tällaista tapahtuu ajanmäärityksen muututtua nykyhetkeen, on viisainta arkistoida käsitellyt ohjelmat jollakin yleisellä arkisto-ohjelmalla, ja asettaa arkiston päivämäärä asianmukaisella tavalla huomioiden se ympäristö, ja paikallinen aika, jossa purku sitten tapahtuu.

-v = laajennettu

Käytettäessä tätä kytkintä **UNP** antaa itsestään lisätietoja. **UNP** ohjelman tekijä lisäsi tämän kytkimen debuggaus- tarkoituksiin.

**[d:][\polku]Syöttötiedosto*

Tiedostomääritystä **UNP** käyttää valitessaan prosessoitavia tiedostoja. Jos annat pelkän **UNP** komennon, mutta et syöttötiedostoa, **UNP** käyttää oletusarvoisesti '*.*' -tiedostonhakumääritystä, eli käsittelee kaikki hakemiston tiedostot. Mutta jos annat syöttötiedoston nimen, käsittelee **UNP** tietenkin vain sen. Jos syöttötiedostosi päättyy '\' merkkiin, **UNP** käyttää '*.*' -valintakriteeriä. Jos taas annetaan pelkkä syöttötiedoston nimi ilman tarkennetta, käytetään valintakriteerinä '.*', ja valitaan tällöin ainoastaan joko **COM** tai **EXE** tiedostoja.

**[[d:][\polku]Tulostustiedosto]*

Tulostustiedoston määrittäminen on lisätoiminto. Jollet määritä sitä lainkaan, pälekirjoitetaan syöttötiedosto. Et voi käyttää tässä yhteydessä jokerimerkkejä. Älä myöskään anna tätä määrittäystä jos aiot käsitellä samalla kertaa useampia tiedostoja.

UNP V3.15 ohjelman tunnistamat tiedostorakenteet

1. Ajettavissa oleva (executable (EXE))

Jos tiedosto alkaa 'MZ' tai 'ZM' signatuurilla, eikä sisällä 'NE' signaturia, otaksutaan, että tiedoston rakenne viittaa nimenomaan **EXE** -tiedostomuotoon, jossa on lisäksi kaksi **UNP V3.15** ohjelman tunnistamaa optiota:

1.1. konvertoitavissa oleva (convertible)

Tiedosto voidaan konvertoida **COM** -tiedostorakenteeksi.

1.2. ladataan ylämuistiin (high memory)

Ohjelma ladataan niin ylös kuin mahdollista varattuun muistilohkoon (tämä vaatii joitakin muita latausrutiineja).

2. Windows tai OS/2 1.x uusi ajettava tiedosto

Tiedosto alkaa 'MZ' tai 'ZM' signatuurilla, sekä sisältää 'NE' -signatuurin.

3. Datatiedosto

Tiedosto ei sisällä 'MZ' tai 'ZM' signaturia, mutta on liian suuri ollakseen **COM** tiedosto (eli sen koko on esimerkiksi reilusti yli 64 kilotavua).

4. Binaarinen (COM)

Tämä näytetään kaikissa muissa tapauksissa.

UNP V3.15 ohjelman viestejä käyttäjälle

Ohjelman tiedotteet käyttäjälle suorittamistaan toimenpiteistä

Tässä luettelo UNP:n antamista tiedotteista käyttäjälle, jotka koskevat sitä, mitä ohjelma on suorittanut:

- decompressing... done

Tässä on suoritettu kenties eniten käytetty UNP -ohjelman toiminnoista.

- removing immunize code... done

Jos tiedosto on *Central Point Anti-Virus* -ohjelman saastuttama, on tiedostoon lisätty koodinpätkä. UNP kykenee poistamaan tämän koodin.

- removing scrambling... done

UNP tunnistaa joitakin 'scrambling' -rutiineita. Jos näet tällaiset viestit, on kyseessä ohjelma jossa on tällainen rutiini.

- removing 'XX' signature

Alkaen UNP V3.02 versiosta voidaan PKlite.EXE ohjelman lisäämä signatuuri poistaa. Tällainen viesti näytetään mikäli UNP on löytänyt poistettavissa olevan signatuurin, ja yrittää poistaa sitä.

- converting to EXE file structure

Tiedosto konvertoidaan yhdeksi EXE -rakenteista

- converting to COM file structure

Tiedosto konvertoidaan yhdeksi COM rakenteista

new size: X

Kun tiedosto on kirjoitettu levyille, UNP ilmoittaa tällä rivillä uuden tiedostokoon.

Ohjelman esittämät kysymykset käyttäjälle

UNP V3.15 ohjelma voi esittää kysymyksiä, sillä vaikka ohjelma tekijä on yrittänyt tehdä UNP ohjelmasta niin fiksun kuin mahdollista, ei se silti osaa lukea ajatuksia, tai tehdä muuta vastaavanlaista. Tästä syystä ohjelma kysyy joskus menettelytapoja.

Add 'pk'/'PK' signature to fake PKLITE decompression (y/n)?

Tämä kysymys esitetään vain jos käytät kytkintä '-k' komentorivillä. Jos vastaat 'Y' tähän kysymykseen, lisää UNP 14 -bitin mittaisen koodin, joka jäljittelee PKlite.EXE tyyppistä dekompressiota. Oikea signatuuri näytetään, ja sitä käytetään automaattisesti. (versiossa V1.20 signatuuri on 'pk' ja muissa 'PK').

File FILENAME.EXT already exists. Overwrite (y/n)?

Tiedostonimi, jota käyttäen **UNP** haluaa kirjoittaa tulostetiedoston on jo olemassa. Jos et ole määrittänyt '-o' -kytkintä, **UNP** kysyy lupaa päällekirjoitukselle. Jos vastaan 'N' ohjelma siirtyy käsittelemään seuraavaa tiedostoa (jos niitä käsitellään useita yhdessä ajossa).

Program is protected, please enter password:

Tämä kysymys esitetään jos yrität laajentaa sellaista ohjelmaa, jota on tiivistetty **TINYPROG.EXE** ohjelmalla käyttäen salasanaa. Salasanaa kysytään, muta sitä ei kysytä siksi, että sillä tarkistettaisiin oletko ohjelman laillinen omistaja, mutta muuta keinoa ongelman ratkaisuun ei juuri ole. Kyseinen ohjelma sisältää useitakin käteviä kytkimiä, joiden avulla ohjelmatiedostoon voidaan jälkikäteen lisätä mm. tekijätietoja.

Remove this routine from file (y/n)?

Olet määrittänyt '-c' -kytkimen, ja **UNP** on löytänyt tiedoston, jota se on havainnut jonkin käsittelevän. Nyt **UNP** kysyy, haluatko poistaa tämän havaitun rutiinin.

-V -kytkintä käytettäessä esitetyt kysymykset

Seuraavat viestit tulevat näytölle vain silloin kun spesifioit '-V' -kytkimen. Ohjelman tekijä on lisännyt nämä debug-tarkoituksia varten.

INFO- Attempting to increase available memory for decompression.

Tämä viesti näkyy vain jos yrität dekompressoida **PKLITE V1.00B(2)** ohjelmalla pakattua tiedostoa. Kun näin tapahtuu, **UNP** käyttää jotakin toista strategiaa laskeakseen muistitilaa jonka se varaa dekompressoitua varten. Oletuksena **UNP** varaa käyttöön niin vähän muistia kuin vain mahdollista. Tämä strategia varaa 15/16 -osaa siitä muistiblokista johon ohjelma on parhaillaan ladattuna.

INFO - command line = " ... "

Tämä viesti näyttää sen kuinka **UNP** on tulkinut asiat, jotka olet kirjoittanut komentoriville. Loistavaa debug-tarkoitusta ajatellen!

INFO - First 64K of base memory has been fully allocated

Jotkut tiivistysohjelmat käyttävät segmenttiä, joka on niiden oman koodin alapuolella. Johtuen siitä, että on mahdollista ladata käyttöjärjestelmä ylämuistiin, tai muutoin yli 640 kilotavun perusmuistitilan, voi olla, ettei saatavissa olekaan täydellistä vapaata segmenttiä. Tämä viesti kertookin, että muistia on varattu sen varmistamiseksi, että alapuolella on täydellinen, vapaa segmentti. Tämä on sama asia minkä **MS-DOS 5.0** järjestelmän **LOADFIX.COM** suorittaa.

INFO - Overlay copy overruled, overlay not copied to destination file

Tavallisesti **UNP** kopioi jokaisen "overlay" -osan, jonka se löytää alkuperäisestä ohjelmasta tulos(kohde)tiedostoon. Tästä on poikkeuksia. Sekä **CRUNCHER.EXE** -että **SEA-AXE.EXE** käyttävät "overlay" -osaa tiedoston tiivistetyn "kopion" säilyttämiseen. Niinpä tämän osan kopiointi aiheuttaisi sen, että ohjelmakoodistosta olisi sen jälkeen olemassa sen kerrannaiset sekä tiivistetyssä -että tiivistämättömässä muodossa Ylläoleva viesti kertookin sen, että "overlay" on poistettu jotta tämänkaltaista ongelmaa ei tulisi.

INFO - Program loaded at XXXXh, largest free memory block: X bytes

Hyvin ilmeistä. Osoite, johon **UNP** latautuu näytetään sen mukaan, mikä on suurin muistiblokki, jonka se voi varata käyttöönsä.

INFO - Unknown program, breakpoints are : GS-XXXX, GI-XXXX, QT-XXXX

Jotkut rutiinit joita **UNP** ohjelman tekijä käyttää ovat hänen mukaansa hieman parempia kuin muut, koska ne (vain) koettavat määrittää tiedostojen "offsettejä" (eli tiedostojen tarkenteita, kuten esimerkiksi **EXE**, ja **COM**) -eivätkä vertaile keskenään niiden signatuureja (esimerkkeinä 'MZ' ja 'ZM'). Jos tämä ovella rutiini löytää 'breakpointeja', mutta ei tunnistamiseen tarvittavia merkkiketjuja, näytetään tämä viesti kertomaan missä nämä tällaiset puutteelliset 'breakpointit' sijaitsevat.

INFO - Using FILENAME.EXT as temp file

UNP kertoo sinulle, että se aikoo käyttää tilapäistä tiedostoa. Tämän tiedoston nimi on yhdistelmä *TEMP*-ympäristömuuttujasta ja tilapäistiedoston nimestä.

INFO - Wildcard matches X filename(s), stored at XXXXh

Tämä viesti kertoo kuinka monta sellaista tiedostonimeä **UNP** on löytänyt jotka vastaavat käytettyä tiedostonimimäärittystä, tai jokerimerkkihakua, sekä sen, mihin nämä nimet on tallennettu.

Ohjelman esittämät varoitukset

Joskus **UNP** suorittaa toimenpiteitä, joista käyttäjän tulisi olla erityisen tarkoin perillä, ja ottaa ne huomioon, ja nämä ilmaistaan varoituksina:

WARNING - Adding 'XX' signature to fake PKLITE decompression

Tässä tapauksessa ohjelma, jota yrität palauttaa takaisin alkuperäiseen kokoonsa on tiivistetty **PKLITE V1.14** ohjelmalla, tai sitä uudemmalla versiolla käyttäen "extra" kompressiosuhdetta [todennäköisesti kyseessä on rekisteröity ohjelmaversio]. Oletuksena **UNP** lisää 14-bitin koodin, joka saa ohjelman "luulemaan" että se on yhä tiivistetyssä tilassa. Jos haluat poistaa TÄMÄN koodinpätkän, voit käyttää kommentoa '*UNP E*'. Yleisesti ottaen "suurimman mahdollisen tiivistyssuhteen" käyttämisestä ei ole vastaavaa hyötyä, koska nykyään levyjen koko on kasvamistaan kasvanut, ja hinnat halpenemistaan halvenneet. Ja aina ne vaan halpenevat, ja halpenevat. Ja yhä vaan halpenevat - no, mitenkä minä nyt tähän juutuinkaan. Jos nyt ohjelmaa tiivistetään kovin runsaalla kädellä, voi vähäininkin vaurio koodissa merkitä toimintakyvyn katoamista, ja virheen korjauksen vaikeutumista, sekä runsasta datan menetystä. Toisaalta jos ohjelmaa siirretään sähköisesti paikasta toiseen, voidaan sitä tällöin siirtää jossakin arkistotiedostossa - esimerkiksi **Zip 2.1** tai **Pkzip 2.04g**:llä luoduissa **ZIP** -tiedostoissa, joita luotaessa voidaan käyttää tiivistystä jotta siirtoaika lyhenisi. Kun **ZIP** -arkistotiedosto taas avataan, palautuu ohjelma, tai muu tiedosto "oikeaan kokoonsa". Jos arkiston siirto epäonnistuu, voidaan aina yrittää uudelleen - ja alkuperäinen ohjelma ei ole muuttunut prosessissa mitenkään.

WARNING - File adds 'XX' signature (added by UNP V3.01 or earlier)

WARNING - File adds 'XX' signature (added by DISLITE V1.15 or higher)

Tässä tapauksessa tiedosto on jo palautettu entiseen kokoonsa, ja siihen on jo lisätty "keinotekoisien" **PKlite.EXE** ohjelman dekompressiosignatuuri. Tämä signatuuri jätetään kuitenkin huomiotta etsittäessä muita mahdollisia dekompressiorutiineja. Alkaen **UNP V3.02** versiosta (sekä **DISLITE**:ssä) ne signatuurit, jotka eivät käytä 'delocation' itemiä, voidaan poistaa komennolla '*UNP E*'.

WARNING - File already has .BAK extension, no backup created

UNP on juuri purkanut tiedoston , ja luonut **BAK** -tiedoston, ja olet (senjälkeen) tehnyt saman uudelleen, sekä spesifioinut '-B' kytkimen. On mahdotonta luoda **BAK** -tiedosto nimelle, jolla on jo ennestään sama nimi, sekä **BAK** tarkenne. **UNP** ylikirjoittaa tässä tapauksessa lähdetiedoston, eikä mitään varmistustiedostoa luoda.

WARNING - File loaded too low in memory to decompress, reloading

Tiedostot, jotka on tiivistetty **EXEPACK.EXE** ohjelmalla vaativat yhden 64 kilotavun segmentin, joka sijaitsee niiden oman koodin alapuolella, jotta ne voitaisiin onnistuneesti palauttaa niiden entiseen kokoon. Jos saatavilla ei ole ainuttakaan täysin tyhjää segmenttiä tähän tarkoitukseen, näyttää **UNP** tämän viestin, sekä lataa tiedoston ylemmäksi muistiin.

WARNING - Infile and Outfile are same, Outfile ignored

Olet spesifioinut tiedoston kahteen kertaan komentorivillä, mikä tarkoittaa sitä, että lähdetiedoston ja tulostustiedoston nimet ovat samanlaiset. Koska tämä on oletusavoinen tilanne, jätetään toinen nimistä huomiotta. Tämä tarkoittaa yksinkertaisesti sanottuna sitä, ettei sinun **TARVITSE** määrittää erikseen tulostustiedoston nimeä jos haluat pitää laajennetulla tiedostolla saman tiedostonimen kuin sillä ennestään oli.

WARNING - Invalid or missing stored header information

Tavallisesti tiivistysohjelma, jota on käytetty tiivistämään käsittelemääsi tiedostoa on säilyttänyt ainakin osan alkuperäisestä tiedoston headerista. **UNP** on vertaillut keskenään tätä tietoa sen tiedon kanssa, jollainen alkuperäisen tiedon tulisi olla, ja **UNP** on tullut tulokseen, etteivät nämä täsmää keskenään. Jos näin käy, käytetään oletusavoista **UNP** headeria.

WARNING - Missing last byte, unable to completely restore file

SHRINK -tiivistysohjelma ei tiivistä oikein tiedostoja, joissa esiintyvät kaikki 256 -merkkiä. Jos tiedostossa esiintyvät kaikki merkit, heitetään viimeinen bitti ohjelmasta pois. Tätä bittiä ei ole enää mahdollista saada takaisin, joten laajennettu ohjelma ei ole yhdenmukainen alkuperäisen kanssa tämän viimeisen bitin (tai useamman) osalta.

WARNING - Outfile specified, -B option ignored

Olet spesifioinut tulostustiedoston ja **'-B'** -kytkimen. Koska **UNP** ohjelman tekijä ei näe tässä mitään järkeä, jätetään **'-B'** -kytkin huomiotta.

Ohjelman viestit jos toimenpidettä ei voitu suorittaa

Joissakin tapauksissa toivottua toimenpidettä ei ole mahdollista suorittaa, tai toimenpide ei onnistunut. Nämä virheviestit kertovat tähän syyn, sekä mitä on tapahtunut. **UNP** jatkaa seuraavan tiedoston käsittelyä:

ERROR - Cannot handle this decompression routine

UNP on tunnistanut tavan, jolla ohjelma on tiivistetty, mutta ei (vielä) osaa palauttaa sitä entiseen kokoonsa.

ERROR - File already is a .COM file

Yrität konvertoida **COM** tiedostoa **COM** tiedostoksi. Tiedäthän toki, että tässä on yhtä paljon järkeä kun Timo Kinnunen nimittäisi itsensä uudelleen Timo Kinnuseksi. Itävaltalaisen **Ludwig Wittgensteinin** sanoin: "se olisi samaa kuin jokin pyörä pyörisi ilman että mikään muu pyörä pyörisi samalla".

ERROR - File already is an .EXE file

Yrität konvertoida **EXE** tiedostoa **EXE** tiedostoksi. Tässä on yhtä paljon järkeä kuin edellisessä.

ERROR - File contains overlay

On hyvä tietää, ettei **COM** tiedostoissa voi olla "overlay" -kerroksia, ja jos haluat konvertoida tiedoston tällaiseksi, poista ensin "overlay".

ERROR - File has invalid entrypoint (CS:IP <> FFF0h:0100h)

Jotta konvertoitu **EXE** tiedosto "alkaisi" oikeasta paikasta, tulisi ohjelman initiaalinen CS:IP osoittaa osoitteeseen FFF0:0100h. Jos näin ei ole laita, saat ylläolevan virheviestin.

ERROR - File has relocation items

Yritit konvertoida **EXE** tiedostoa **COM** tiedostoksi niin, että sillä on 'relocation' itemsit. Kuitenkaan **COM** tiedosto ei osaa käsitellä tällaisia.

ERROR - File is too large for .COM file

COM tiedoston suurin mahdollinen koko on paljon pienempi kuin **EXE** tiedoston. Niinpä voi hyvinkin sattua, että **EXE** on liian suuri tullakseen konvertoiduksi **COM** tiedostoksi.

ERROR - Unexpected call to INT 21h, decompression failed

Kun **UNP** lajentaa tiedostoja takaisin entiseen kokoonsa, se siirtää kontrollin ohjelmalle. Jos se ei saa kontrollia takaisin, on hyvin todennäköistä, että 'keskeytys 21h' tapahtuu ennemmin tai myöhemmin (*INT 21h* on kaikkein tärkein keskeytys). **UNP** tarkistaa odottamattomat kutsut tähän keskeytykseen varmistaakseen, että sillä on yhä kontrolli. Voit ohittaa tämän tarkistuksen '-i' -kytkimellä.

Viestit "fataalisen" virheen tapauksessa

Kun näytölle ilmaantuu jokin seuraavista viesteistä, on jotakin perustavalla tavalla pielessä, ja **UNP** ei pysty jatkamaan työskentelyään. Se todennäköisesti palaa käyttöjärjestelmätasolle ja antaa *EXIT* koodin.

FATAL ERROR - Decompressing many files into one

Olet spesifioinut lähdetiedoston, mutta ohjelma löytääkin useamman kuin yhden samannimisen tiedoston. Vanhassa *FAT* -tiedostojärjestelmän **MS-DOS** järjestelmässä, jossa kaikki tiedostot ilmaistaan suurin kirjaimin, ja siinä on mahdotonta antaa samaa tiedostonimeä ohjelmille samassa hakemistossa, ei tiedostonimien virheellisestä tulkinnasta ole tavallisesti vaaraa.

FATAL ERROR - Divide overflow (INT 00h) generated by CPU

Tämä merkitsee sitä, että on ajettu virheellinen *DIV*-instruktio. Tavallisesti DOS pysäyttää tässä tapauksessa ohjelman. **UNP** menettelee tässä kohden niin, että se asettaa keskeytysosoittimet takaisin ennenkuin se lopettaa toimintansa.

FATAL ERROR - No files found matching FILENAME.EXT

UNP ei löytänyt yhtään dekompressoitavaa tiedostoa.

FATAL ERROR - Not enough memory ...

UNP yritti varata muistia, mutta sai virhepalautuksen. Tämä viesti kertoo ettei muistia ole käytettävissä toimintoon.

FATAL ERROR - Output path/file must not contain '' or '?'*

Olet käyttänyt jokerimerkkejä tulostustiedoston nimessä, mikä ei ole sallittua.

FATAL ERROR - User abort, ^C/^Break pressed (INT 23h)

Keskeytystä 23h kutsutaan jos DOS havaitse, että näppäimiä 'Ctrl-C' tai 'Ctrl-Break' on painettu. **UNP** lukitsee tämän kyetäkseen palauttamaan käyttämänsä keskeytykset.

UNP V3.15 ohjelmalla poistettavat rutiinit

Haluat ehkä tietää, mitä ohjelmia **UNP V3.15** voi käsitellä. Seuraavassa luettelo rutineista, joita ohjelma ainakin kykenee poistamaan.

COM tiedostoista löytyvät rutiinit

CENTRAL POINT ANTI-VIRUS V1 ; immunize code

COMPACT V4.4

COMPACT V4.5

DIET V1.00

DIET V1.02b or V1.10a

DIET V1.20

ICE V1.00

PKLITE V1.00β

PKLITE V1.03

PKLITE V1.05

PKLITE V1.12

PKLITE V1.13

PKLITE V1.14

PKLITE V1.15

PROTECT! COM/EXE V1.0 or V1.1

PROTECT! COM/EXE V2.0

PROTECT! COM/EXE V3.0

PROTECT! COM/EXE V3.1

PRO-PACK V2.08

PRO-PACK V2.08

PRO-PACK V2.08

PRO-PACK V2.08

SCRNCH V1.00

SHRINK V1.00

EXE tiedostoista löytyvät rutiinit

CENTRAL POINT ANTI-VIRUS V1; immunize code

COMPACT V4.4

COMPACT V4.5

CRUNCHER V1.0; +.COM files

DIET V1.01

DIET V1.00d ; small & large, with & without items

DIET V1.02b, V1.10a or V1.20 ; small & large, with & without items

DIET V1.44; small, large + .COM files

DIET V1.44, choose great SFX routine ; small, large + .COM files

DIET V1.45f ; small, large + .COM files

DIET V1.45f choose great SFX routine ; small, large + .COM files

DISLITE V1.15 [tai uudempi] ; small signatures (no items in sig)

EXEPACK V4.00
EXEPACK V4.05 or V4.06
EXEPACK patched with EXPAKFIX V1.0
LINK /EXEPACK V3.60 or V3.64
LINK /EXEPACK V5.31.009
KVETCH V1.02B
LZEXE V0.90
LZEXE V0.91 or V1.00a
PGMPAK V0.14
PKLITE V1.00B 0,1,2
PKLITE V1.03 0,1,2,3
PKLITE V1.05 0,1,2
PKLITE V1.10 3
PKLITE V1.12 0,1,2,3
PKLITE V1.13 0,1,2,3
PKLITE V1.14 0,1,2,3
PKLITE V1.15 0,1,2,3
PKLITE V1.20 1
PKLITE V1.20 1,3
PRO-PACK V2.08
PRO-PACK V2.08
PRO-PACK V2.08
PRO-PACK V2.08
PROTECT! COM/EXE V1.0
PROTECT! COM/EXE V1.1
PROTECT! COM/EXE V2.0
PROTECT! COM/EXE V3.0
PROTECT! COM/EXE V3.1
SEA-AXE
TINYPROG V1.0
TINYPROG V3.3
TINYPROG V3.6
UNP V3.02 [tai uudempi]; fake PKLITE signature

ja niin edelleen...

LZEXE V1.00a, PGMPAK V0.14; PKLITE, SCRUCH, SHRINK V1.00

LZEXE V1.00a:

Tämä ohjelma on sama kuin **LZEXE V0.91 (Bellardin valmistama ilmaisversio)**, lukuottamatta jotakin pientä. Tätä ohjelmaa tarjottu INFO PC:n lukijoille (IS2 *France Diffusion*)

PGMPAK V0.14:

Kun tiedosto tiivistetään tällä ohjelmalla, lisätään tulostustiedostoon 12-bitin suuruinen "lisäkerros". Se, mitä lisätään on nimi ja versionumero ("PGMPAK V0.14"). Tätä "kerrosta" ei poisteta automaattisesti kun palautat tiedoston alkuperäiseen kokoonsa, vaan sinun täytyy käyttää '-R' -kytkintä.

PKLITE:

PKLITE V1.00B näyttää sisältävän virheen. Joitakin tiedostoja sillä käsiteltäessä niihin lisättiin 512-bittiä. On lienee tarpeetonta sanoa, että tämän jälkeen tiedostot eivät enää toimineet normaalisti. Koska

UNP kirjoittaa tiedostot takaisin sellaisina kuin ne toimisivat muistissa, niin eivät nämä tällaiset **UNP** ohjelmallakaan alkuperäiseen kokoonsa laajennetut tiedostotkaan toimi. Jos kuitenkin käytetään tätä **PKLITE** -ohjelmaa, ja annetaan komento:

PKLITE -X

Saadaan tulokseksi alkuperäinen tiedosto. Siis: käytettäköön tätä menetelmää tässä tapauksessa.

PKLITE V1.14 (tai uudemmat)

Ainakin dokumentaation mukaan - lisäävät signatuurin 'PK' "extra"-tiivistettyihin tiedostoihin, jotta ohjelmat voisivat itse tarkistaa "ovatko ne yhä vielä **PKLITE** ohjelmalla tiivistetyssä tilassaan". Jotta **UNP** ohjelman avulla entiseen tilaansa palautettuja ohjelmia voidaan estää havaitsemasta tätä muuttunutta tilaansa, **UNP** lisää tulostus- tiedostoihin 14-bitin koodin, joka korvaa signatuurin PSP:ssä samalla tavoin kuin **PKLITE** tekee.

SCRUNCH:

Useimmissa tapauksissa **UNP** ohjelman dekompressorutiinit on luotu testitiedostoilla. **UNP** ohjelman kirjoittajalla ei ollut käytettävissään **SCRUNCH** ohjelmaa, joten sen toiminnan kartoitus ei onnistunut. Tietenkin on olemassa muita samanlaisia rutiineja. **UNP** ohjelman tekijällä oli kuitenkin käytössään **SCRUNCH** ohjelmalla tiivistetty tiedosto, joka oli konvertoitu **COM** tiedostoksi ennen tiivistystä. Tiedosto sisälsi nimittäin 'relocation' itemsejä (joita **EXE** tiedostot käyttävät). Tämä mainittu konversio voi hyvinkin olla osa **SCRUNCH** ohjelman suorittamaa tiivistystä ...

SHRINK V1.00:

Tämä tiivistysohjelma on *Run Length* -kompression huono sovellus, sillä se sisältää kaksi virhettä, joista toinen on nimenomaan tiivistysrutiinissa. Virheet tulevat esiin kun tiivistetään tiedostoa, joka sisältää kaikki 256-bittiä. **UNP** ohjelman tekijä on kirjoittanut tätä tiivistysohjelmaa varten rutiinin, joka ratkaisee ensimmäisen virheitä, mutta ei toista, jossa viimeinen bitti jätetään tiedostosta kokonaan pois. Kun näin käy, ei tiedostoa voida rakentaa enää entiselleen. Tästä huolimatta dekompressointi **UNP** ohjelmalla kannattaa aina - vaikka viimeinen bitti jääkin puuttumaan.

Miten toimivat Lzexe v.0.91 (β-release) ja PKlite 2.50 sekä eräät sähköisten julkaisujen tekon tarkoitetut ohjelmat

```
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX
Z:\>C:
C:\>LZEXE.EXE
LZEXE.EXE Version 0.91 (c) 1989 Fabrice BELLARD
Compacteur à hautes performances de fichiers EXE

Syntaxe: LZEXE nomfichier[.EXE] [chemin]
"nomfichier" est le nom du fichier EXE que vous voulez compacter.
L'extension EXE est ajoutée si vous l'omettez.
Le fichier compacté est créé dans le répertoire courant.

Pour plus d'infos voir la documentation: LZEXE.DOC

Ce programme a été conçu et réalisé par Fabrice BELLARD qui a décidé
généreusement de le mettre en FREEMWARE.
Si vous avez des problèmes ou des suggestions pour ce programme,
vous pouvez écrire à:      Fabrice BELLARD
                              451 chemin du mas de matour
                              34790 GRABELS (FRANCE)

ou envoyer un message en BAL sur RTEL ou QBSS au pseudo: FAB

C:\>_
```

Fabrice Bellardin tekemää **Lzexe.EXE** ohjelmaa käytetään DOS -käyttöjärjestelmässä toimivien **EXE** tiedostojen tiivistämiseen, ja erityisen sopiva se on käytettäväksi tiettyjen tällaiseen muotoon tehtyjen *sähköisten julkaisujen* tiivistyskäsittelyyn, jolloin lopputulos voi vastata kooltaan alkuperäistä tekstitiedostoa. **Lzexe** kompressoit kuitenkin pelkästään **EXE** -tiedostoja, eli ohjelmana ajettavissa olevia tiedostoja. Tämän sovelluksen lisäksi on olemassa muitakin hyviä kompressoreja, kuten erinomaiset **PKlite.EXE**, **TINYPROG.EXE**, ja **LHARC.EXE**. Kaikki mainitut ohjelmat toimivat samaan tapaan - eli ne koodaavat tiedostoissa usein toistuvia ketjuja lyhyempään muotoon, jonka purkamista varten ne asettavat agentin, joka suorittaa tämän laajennusoperaation kun tiedosto käynnistetään. Lähes kaikissa käsitellyissä **EXE** -tiedostoissa dekompressointiaika on niin lyhyt että sillä ei ole käytännön merkitystä. Lisäksi dekompressointi ei vaadi levyiltä lisätilaa, tai lisätilaa RAM-muistista. **Lzexe.EXE** yksinkertaisesti käyttää sen määrän RAM-muistia, joka tavallisesti varataan kompressoimattomille tiedostoille. Kompressointialgoritmi on optimoitu niin nopeuden kuin tehokkuudenkin puolesta, eli **EXE** -tiedostot kompressoitään lähes siihen kokoon johon **ZIP** -tiedostot puristetaan. Helpoin tapa luoda sähköinen DOS -julkaisu on kirjoittaa tavallinen **ASCII** -tekstitiedosto, ja käyttää sen esittämiseen pientä ohjelmaa, joka näyttää sen sisällön sellaisenaan, ja jolla tekstitiedostoa voidaan selata. Tässä tapauksessa tekstitiedostoa ei tiivistetä mitenkään mitenkään, ja tarvittaessa sen tietojen "päivittäminen" on helppoa. Ainut käyttörajoitus on se, että tekstitiedoston nimen täytyy aina olla sama kuin mitä ohjelmaosioon on asetettu - tai muussa tapauksessa täytyy kirjoittaa "uusi" esitysohjelma, tai varustaa se tiedostojen käsittelyn ja valinnan mahdollistavalla systeemillä. Tässä tapauksessa **Lzexe.EXE** ohjelman käyttö ei ole tarpeen, koska esittävä ohjelmaosio on muutoinkin jo kooltaan varsin pieni, ja koska varsinaista esitettävää tekstiosiota ei ole tarkoitettu lainkaan tiivistettäväksi. Tämantapainen järjestely on esimerkiksi *MouseWare 7.21* paketin mukana toimitetussa **Readme.EXE** ja **Readme.txt** -parissa, joista edellinen esittää jälkimmäisen DOS -tilassa. **Lzexe.EXE** -ohjelmaa kannattaa käyttää erityisesti sellaisten sähköisten julkaisujen tiivistämiseen, joissa käynnistävä osa on kiinteä osa esitettävää tiedostoa, ja ne on yhdistetty yhdeksi tiedostoksi.

Lzexe.EXE ohjelmaa käyttävät oletuksena käsittelemiensä tiedostojen tiivistämiseen sellaiset ohjelmat kuten **BigText.EXE** ja **TLife.EXE**, jotka toimivat yhdistämällä tekstiin käynnistävän, ja esittävän osion, ja joita käyttämällä voidaan luoda hakemisto, josta voidaan siirtyä suoraan valittuun tekstinosaan. Tiivistämättömän tiedoston koko on kuitenkin melkein kaksinkertainen alkuperäiseen tekstitiedostoon

verrattuna, ja siksi sen tiivistys kannattaa. Molemmat mainituista sovelluksista lisäävät jokaiseen luomaasi tiedostoon ohjelman tekijää koskevia tietoja - ja jos sovellukset halutaan muuttaa sellaisiksi, että ne automaattisesti asettavat sinun tietosi luotaviin sähköisiin julkaisuihin, pitää **BigText.EXE** ja **TLife.EXE** -ohjelmat laajentaa esimerkiksi **UNP** -ohjelmalla takaisin "alkuperäiseen" kokoonsa, ja sen jälkeen jollakin levyeditorilla, kuten **Norton Utilizer 4.5** pakettiin kuuluvalla **NU.EXE** ohjelmalla muuttaa tiedot vastaamaan omia tietoja. Kuitenkaan en pitäisi tätä enää vaivan arvoisena, koska tällaisilla sähköisillä julkaisuilla tuskin enää on kysyntää. Pohjana **TLife.EXE** ja **BigText.EXE** -sovelluksilla suoritettavalle käsittelylle on aina puhdas *ASCII* -tekstitiedosto, ja tuloksena elektronisia kirjoja, jotka itsenäisesti näyttävät sisältönsä DOS -tilassa ajettuina, ja ne käynnistetään samalla tavoin kuin muutkin ohjelmatiedostot. Näiden "ohjelmatiedostojen" puutteena on se, että niiden sisältämän tiedon päivittäminen helposti ei ole mahdollista muutoin kuin säilyttämällä ne tekstitiedostot, joita on käytetty, ja päivittämällä tietoja niihin, ja luomalla tarpeen mukaan niistä uusia esityksiä. Toisaalta niihin saa helposti liitettyä toiminnot, joilla esimerkiksi tulostus tiedostoon, tai kirjoittimelle on mahdollista, ja siten vastaanottaja saa niin halutessaan aina *ASCII* tekstitiedoston samalla kun saa esitysohjelmankin. Tulostiedostojen toiminta tehostuu selvästi DOS -järjestelmässä, ja ainakin pienillä kiintolevyillä - jos esitystiedostot tiivistetään. Periaatteessahan ne ovat edelleenkin tavanomaisia tekstitiedostoja, vaikka niissä onkin **EXE** -ohjelmaosa, ja purkukoodiosa, ja vaikka niihin on luotu tavallisesti yksinkertaisia hypertekstiominaisuuksia, kuten tiedoston alussa oleva sisällysluettelo, josta voidaan siirtyä haluttuihin lukuihin suoraan enteriä painamalla. Ennenkuin teksti muunnetaan **BigText.EXE** ohjelmalla, saadaan tällainen toiminto aikaan merkitsemällä kunkin luvun aloituskohdat tekstissä "tilde" -merkillä, ja luomalla vastaavan määrän otsikoita tiedoston alkuun ja koodaamalla ne kaikki /M: -koodilla. Tavallisesti tiedostoon syntyy tiivistyksen yhteydessä myös "Info" -optio, jossa on tietoja prosessissa käytetystä ohjelmasta, ja sen tekijästä. Juuri tämän informaation muuttamisesta **NU.EXE** ohjelmalla puhuimme aikaisemmin. **BigText.EXE** ohjelma soveltuu erityisen hyvin suurempien tekstitiedostojen käsittelyyn, mutta sillä on tietty yläraja tässä suhteessa - eli muutaman sadan kilotavun tienoilla. Määrä selviää tavallisesti kokeilemalla, mutta on selvää, että mitään suurta teosta tällä tavoin ei voida luoda. Ellei haluta käyttää **Lzexe.EXE** ohjelmaa tiivistykseen, on hyvä valinta myös **TINYPROG.EXE**, jota käyttäen tiedostoihin voi liittää monia erityispiirteitä, kuten omaa persoonaa koskevia tietoja, jotka eivät näy julkaisun lukijalle, mutta jotka ovat selvästi nähtävissä jollakin levyeditorilla tiedoston alussa. Hyvä valinta on myös aiemmin mainittu **Pklite.EXE**. Jos **BigText.EXE** -ohjelmalla tehty tiivistämätön **EXE** tiivistetään **TINYPROG.EXE** ohjelmalla, ei sen sisältö ole enää luettavissa edes levyeditoreilla muutoin kuin koodeina. Itse asiassa tiedosto on tavallaan tehokkaasti "salakirjoitettu". Valitettavasti tässä on se varjopuoli, että mikäli tiivistettävässä ohjelmassa on jokin tahallisesti istutettu virus, myös sen "selväkieliset" viestit peittyvät, ja virusskannerit eivät virusta löydä. Onneksi tällainenkin tiivistetty tiedosto voidaan palauttaa alkuperäiseen kokoonsa useimmiten **UNP V3.15** -ohjelmalla, jolloin sen sisältämä tekstikin on vielä luettavissa ainakin levyeditorilla, ja siten myös "pelastettavissa" muiden sovellusten käsittelyyn. Ellei tekstiä saa muutoin pelastettua talteen, voidaan sen alku- ja loppukohta merkitä tällaisesta tiedostosta esimerkiksi **List.EXE** -ohjelmalla, ja tallentaa halutulle tiedostonimelle, ja käsitellä sen jälkeen jossakin videoeditorissa. Teoriassa tässäkin operaatioissa on tietty riskinsä. Jos aiot luoda vielä nykyäänkin tällaisia **EXE** -esitystiedostoja **BigText.EXE** ohjelman tapaisilla sovelluksilla, kannattaa tiedostoihin aina liittää optiot, jotka mahdollistavat sisällön tulostamisen tiedostoon, tai ainakin kirjoittimelle.

Lzexe.EXE -ohjelma on julkista omaisuutta, ja voit tästä syystä käyttää, kopioida, ja jakaa sitä vapaasti. Voit myös käyttää sovellusta kaupalliseen tarkoitukseen - eli voit käyttää sitä esimerkiksi niiden ohjelmien valmistukseen, joita aiot myydä. **Lzexe.EXE** -ohjelman itsensä myyminen ei sensijaan ole sallittua. Ohjelman vaatimukset eivät ole suuret, ja sille riittää koneeksi perus PC, ja vastaavat, joissa on 80286 tai 80386 -suorittimet. Muistivaatimukseen ei ole mahdoton, koska 128 kilotavua pitäisi riittää. Ohjelman käyttö on yksinkertaista, koska ainut mitä tarvitaan on DOS -komentokehoitteesta annettu:

Lzexe tiedostonimi

jossa "tiedostonimi" on kompressoitavan tiedoston nimi. Tarkennetta ei tarvitse määrittellä, koska **EXE** -tunniste liitetään luotavaan tiedostoon oletusarvona. "Polku" on lisämäärite, ja se viittaa siihen hakemistoon, jossa tiivistettävä tiedosto ja työtiedosto **LZTMP.EXE** sijaitsevat. Huomaa kuitenkin, että jotkut tiedostot ovat **EXE** -tiedostoja ainoastaan näennäisesti. Tästä syystä DOS ei tunnista **EXE**

-tiedostoja pelkäästään niiden tiedostotarkenteen perusteella, vaan siitä, onko tiedoston "*header*" -osassa kirjaimet "*MZ*", jota seuraavat tiedot koskien tiedoston RAM -muistissa varaamaa tilaa, jne. Niinpä jos pelkäästään muutat esimerkiksi **COM** tarkentimen **EXE** -tarkentimeksi, **Lzexe.EXE** ei käsittele tällaista tiedostoa. Kuitenkin ohjelma voidaan "pakottaa" hyväksymään myös **COM** -tiedosto, jos se käsitellään **ComToExe.EXE** ohjelmalla. Jos nimittäin satut sellaisen vielä mistään löytämään. Se toimii kuin DOS järjestelmän **EXE2BIN.EXE** ohjelma, mutta päinvastaiseen suuntaan. Jotkut aikaisemmista krakkeriohjelmista, kuten **Copywrit.EXE**, kykenivät monistamaan "kopiointisuojaattuja" levykkeitä. Esimerkiksi jokin sovellusohjelma, kuten DOS -pohjainen **Lotus 1-2-3** oli suojattu kopiointilta niin, että ennen sen käynnistystä "laillisuus" varmistettiin tietystä tiedostosta, joka oli kiintolevyllä. **Copywrit.EXE** kykeni käsittelemään sen käynnistävää **COM** -ohjelmaosiota niin, että siitä tuli **EXE**, ja kopiointitarkistus jäi pois. Nykyäänhän tälläkään tiedolla ei ole merkitystä, koska **Lotus 1-2-3** -sovellusta ei juurikaan käytetä. Aikaisemmin valtaosa DOS -peleistä myytiin levykkeillä, ja niissä oli useita erilaisia kopiointin estäviä rakenteita. **Copywrit.EXE** oli tarkoitettu lähinnä tällaisten levykkeiden monistamiseen. Turvallisuuden lisäämiseksi **Lzexe.EXE** ei tuhoa alkuperäistä tiedostoa, vaan nimeää sen tunnusosalla **OLD**. Tämän lisäksi ohjelma luo prosessin aikana tilapäis tiedoston **Lzexe.TMP**, joka nimetään akuperäisen ohjelman nimelle vasta kompressoitiprosessin täysin päättyttyä.

Millaisia tiedostoja **LZexe** ei osaa käsitellä?

Tiettyjä tiedostoja **Lzexe.EXE** ei voi kompressoida seuraavista syistä:

1. Jos tiedosto ei ole aito **EXE** tiedosto. Tähän hyvä ratkaisu on käyttää **ComToExe** -ohjelmaa.
2. Jos **EXE** ohjelman "*relocation*" -taulu on liian suuri. **EXE** ohjelmatiedosto voi koostua useasta segmentistä - eli ohjelma on joukko toimivia osakokonaisuuksia - ja tämä ei ole **COM** -tiedostossa mahdollista. Tästä syystä **EXE** tiedoston on sisällettävä taulu, jossa erikseen määritellään mihin kohtaan muistia ohjelma asettuu, ja mihin osoitteisiin se voi kutsua aliohjelmaa, koska muutoin ohjelma ei voisi toimia. Tämä tarkoittaa sitä, että **EXE** ohjelma suorittaa itsenäistä muistinhallintaa, ja tästä syystä näitä ohjelmia ei oikeastaan ole viisasta ajaa muussa kuin aidossa DOS -tilassa, tai sellaisessa jonkin järjestelmän DOS -ikkunassa, joka on mahdollisimman hyvin suojattu, ja joka jäljittellee mahdollisimman tarkoin aitoa DOS -tilaa. Tämä siitä syystä, että DOS järjestelmässä itse järjestelmä luo vain puitteet, eli se aktivoi erilaiset muistityypit, ja niiden määrän, joita sovellukset pyrkivät sitten itsenäisesti hallitsemaan. Jos muistinkäytössä tapahtuu konflikteja eri sovellusten välillä, koko DOS järjestelmä kaatuu.
3. Jos käsiteltävä **EXE** ohjelma itsessään on hyvin suuri, se voi myös sekaannuttaa **Lzexe.EXE** ohjelman, joka luo taulun, jossa on 16,000 "*relocation*"-osoitetta, mikä riittää lähes kaikia **EXE** tiedostoja varten, mutta poikkeuksiaahan aina on.
4. Joskus tiedosto, jota yrität kompressoida, voi jo olla kompressoitu **Lzexe.EXE** ohjelmalla, tai jollakin muulla tiivistysohjelmalla. Huomaa, että voit kompressoida **Lzexe.EXE** ohjelmalla edelleen esimerkiksi tiedostoja, jotka on jo ennestään tiivistetty jollakin toisella sovelluksella, mutta joilla tiivistys on tapahtunut tehottomasti. Esimerkiksi Microsoftin **EXEPACK.EXE** ei ole yhtä tehokas kuin **Lzexe.EXE**, ja suurempi tiivistysaste voidaan saavuttaa purkamalla tiivistetty tiedosto ensin **UPACKEXE.EXE** -ohjelmalla, joka kumoaa **EXEPACK.EXE** ohjelman vaikutuksen. Kun tiedosto tiivistetään tämän jälkeen **Lzexe.EXE** ohjelmalla, on sen koko jo huomattavasti pienempi.
5. Joskus kompressointi ei tuota merkittävää hyötyä -kuten jos käsitellään hyvin pieniä tiedostoja (pienempiä kuin 2 Kilobittia).

Tiivistetyistä ohjelmista aiheutuvia vakavampia ongelmia, kun on käytetty **Lzexe.EXE** tiivistysohjelmaa

Ongelmia aiheuttavat tietyt kompressoitujen **EXE** tiedostot voivat lukkiuttaa järjestelmän. Näin saattaa

käydä jos ohjelma tarkistaa aina käynnistyessään kokonsa (kuten *Turbo Debugger*), tai jos ohjelma tarkistaa ennen käynnistymistään sijaintinsa levyiltä. Ongelmia voi seurata myös silloin kun ohjelmassa on "overlay"-elementtejä, jotka ladataan muistiin myöhemmin kuin pääosa ohjelmasta, ja joilla on tästä syystä oltava levyllä kiinteä sijainti, jotta ne löytyisivät varmasti. **Lzexe.EXE** varoittaa näistä yleensä etukäteen. Huomaa, että monissa uudemmissa käyttöjärjestelmissä ohjelmia ei enää suunnitella niin, että niiden toiminta perustuisi niiden itse määrittelemiin muistinvarauksiin, tai tiettyihin levysektorisijainteihin. Tämä ei kuitenkaan tarkoita sitä, ettekö voisi DOS ohjelmia lainkaan ajaa esimerkiksi **Windows 95** järjestelmässä, vaan sitä, että niiden mahdolliset tuhoisat vaikutukset tulisi tarkkaan ennakoida. Ongelmia voivat tuottaa myös ohjelmat, jotka vaativat *Microsoft Windowsia*. Nämä ohjelmat eivät ole todellisia **EXE** tiedostoja, ja niitä ei voi kompressoida menestyksellisesti **Lzexe.EXE** ohjelmalla. **PKlite.EXE** ohjelman 2.50 -versio kykenee tiivistämään *Windows 3.0* tiedostoja, mutta ei enää kykene palauttamaan niitä takaisin alkuperäiseen kokoonsa. **Windows 3.1** käyttöjärjestelmän ajonaikaisia **DLL** -kirjastotiedostoja ei voi tiivistää esimerkiksi **PkLite.EXE** -ohjelmalla. Huomaa, että tietyt ohjelmat käyttävät konfigurointioptioita, jotka muotoilevat koodia (kuten esimerkiksi *Turbo Pascal*). Tällaisessa tapauksessa - konfiguroi ensin ohjelma, ja kompressoi se vasta sitten. Jos teet näin, ota alkuperäisestä aina varmuuskopio.

Eräitä **Lzexe.EXE** ohjelman tiivistysmenetelmää koskevia teknisiä huomioita

Kompressointialgoritmi perustuu *Ziv Lempel* -menetelmään, ja se käyttää sirkulaarisia (ring) puskureita, ja *tree-lile* -menetelmää löytääkseen byttisekvensien toistoja. Sijainnin enkoodaus, ja toistuvien sekvenssien pituudet optimoidaan apualgoritmeilla, joka perustuu *Huffmannin* menetelmään. Kompressoimattomat bitit pidetään muuttumattomina, koska mikä tahansa edelleentyöstämismenetelmä (kuten *Adaptiivinen Huffman LHARC.EXE* -ohjelmassa, tai *Shanon-Fano puut PKzip.EXE* -ohjelmassa) voisi pidentää dekompressioaikaa, ja ennen kaikkea, koska vaadittaisiin monimutkaisempi ja suurempi dekompressor, joka vähentäisi huomattavasti kompressiosuhdetta.

Dekompressor sijaitsee **EXE** tiedoston lopussa, ja sen koko on 330 bittiä. Sen toiminnot ovat:

1. Tehdä *Cyclic Redundancy Check* -tarkistus sen varmistamiseksi, ettei tiedostoa ole käsitelty -mikä on hyödyllinen ainakin sellaisten virusten varalta, jotka kirjoittavat jotakin tiedostoon lisäten samalla sen kokoa. *Cyclic Redundancy* -testi suoritetaan vain dekompression aikana, mutta mitään viestejä se ei anna virhetilanteessa.
2. Asentaa itsensä korkealle RAM-muistiin, ja siirtää sen jälkeen kompressoitu koodi jotta jäisi tilaa **EXE** tiedostolle.
3. Dekompressoida koodi, ja tarkistaa että se on virheetön, ja säätää segmentit, jos tiedoston koko on enemmän kuin 64 kilotavua.
4. Dekompressoida "relocation" -taulu, ja päivittää uudelleensijoitettavissa olevat **EXE** tiedoston osoitteet.
5. Ajaa ohjelma, ja päivittää CS, IP, SS, ja SP -rekisterit.

Tämä on kaikki on suuremmissa, jos haluat säästää levytilaa, tai jos sinulla on sitä käytettävissä hyvin vähän, mutta menetelmällä on myös varjopuolensa. Kuka tahansa voi nyt levittää viruksia - ei tarvita muuta kuin "tartutetaan" ohjelmatiedosto, ja sen jälkeen tiivistetään tiedostoa fyysisesti pienemmäksi, niin virus tulee näkymättömäksi (sekä viesti, josta jotkut viruksentorjuntaohjelmat tunnistavat saastuneita tiedostoja). Niinpä niin. Tänä päivänä riski ei ole kuitenkaan kovinkaan suuri, koska käytetään sentapaisia käyttöjärjestelmiä, ja ohjelmia, jotka eivät ole DOS -sovelluksia, mutta eihän virus silti häviä sellaisesta tiedostosta, johon se on tartutettu, ja jota torjuntaohjelmat eivät ole paikantaneet. Toisaalta: Harvoinpa tavallisella tietokoneenkäyttäjällä on varastossaan toimivia viruksia, joita hän sitten vahingoniloinen virne suupielessään istuttaa omatekoisiin ohjelmiinsa, ja joita hän sitten toiveikkaana jakaa muille - pahaa-aavistamattomille ystävilleen. Jos koneessa on viruksia, ne voivat siirtyä ystäville aivan

tahattomastikin, ja ilman mitään kavalia salaustoimenpiteitäkin!! Se, miksi virusviestit tulevat näkymättömiksi johtuu samasta syystä mistä itse tiivistettävän ohjelman omat, sisäiset viestitkin käyvät näkymättömiksi. Teksti on aina kaikkein helpoimmin tiivistettävä osa mitä tahansa ohjelmaa, koska se koostuu toistuvista kirjainyhdistelmistä, ja niitä voidaan aika helposti koodata suuria määriä käyttämällä lyhyempää, niitä kuvastavaa ilmaisua. Jos hankkii pelkästään kaupallisia ohjelmia, voidaan ainakin myyjä voidaan asettaa vastuuseen vahingoista helpommin kuin jos jokin suosittu ohjelma olisi ostettu esimerkiksi ulkomailta "halvalla", ja käsitelty tällä tavalla. Sama pätee myös tiivistettyihin sähköisiin julkaisuihin, jotka sisältävät etupäässä tekstiä, ja jotka on muovattu **EXE** -muotoon.

Loppuarvio ja loppulaulu *Cadillacin* kunniaksi

UNP V3.15 - ja muut tässä esittelemäni ohjelmat näyttäsivät olevan melko käyttökelpoisia, ja **UNP** silloin jos haluaa avata sellaisia ohjelmätiedostoja, jotka on jossakin salailutarkoituksessa tiivistetty pienemmäksi. mutta joissa ei ole varsinaisia salausrutiineja. Niihin on senjälkeen helppoa kirjoittaa jollakin editorilla mitä haluaa, ja ne voi senjälkeen taas tiivistää takaisin alkuperäiseen ahtauteensa. En yllytä ketään, enkä edes toivo kenenkään tekevän tällaista. Mutta ohjelman mahdollisuudet on selkeästi sanottava. Mutta lopuksi haluaisin "julkaista" eräänlaisena keventävänä reikänä tässä maailmassa suomennoksen eräästä laulusta: **Kätilläk** - laulu. lauletaan lähinnä vaarivainaan nuotilla, ja verkkaisesti samalla kitaraa rämpyttäen - minkä täytyy tapahtua selvästi hitaammin kuin mitä itse laulun poljento etenee:

"missä on lapseni poissajama kätilläk"
"missä on lapseni poissajama kätilläk"
"milloinkaan eet taakaasiin tule säkk"
"ooi laapsi, laapsi laapsi, oolehyyvä"
"VAUUUU"

[ja sitten ontuva, ja vinkuva kitarasoolo]

Alunperin esittänyt englanninkielellä (Suomessakin pissapottushowtaan esittämässä käynyt englantilaisyhtye) **Renegades**