

Εισαγωγή στο NanoBSD

Daniel Gerzo

Κυρί'ακος Κεντρωτής

\$FreeBSD: release/9.2.0/el_GR.ISO8859-7/articles/nanobsd/article.xml 41645
2013-05-17 18:49:52Z gabor \$

Διάδιαδότη Άέέάέπιάόά © 2006, 2009 Ε Ομάδα Τεκμηρίωσης του FreeBSD
\$FreeBSD: release/9.2.0/el_GR.ISO8859-7/articles/nanobsd/article.xml 41645
2013-05-17 18:49:52Z gabor \$

Το **FreeBSD** είναι ένα κατοχυρωμένο εμπορικό σύμβολο του **FreeBSD Foundation**. Πολλές από τις λέξεις ή φράσεις οι οποίες χρησιμοποιούνται από τους κατασκευαστές ή τους πωλητές τους για να διακρίνουν τα προϊόντα τους θεωρούνται εμπορικά σύμβολα. Όπου αυτές εμφανίζονται σε αυτό το κείμενο και για όσες από αυτές γνωρίζει η Ομάδα Ανάπτυξης του **FreeBSD** ότι είναι πιθανόν να είναι εμπορικά σύμβολα, θα δείτε ένα από τα σύμβολα: “™” ή “®”.

Το έγγραφο αυτό παρέχει πληροφορίες για τα εργαλεία του **NanoBSD**, τα οποία μπορούν να χρησιμοποιηθούν προκειμένου να δημιουργηθούν είδωλα του συστήματος **FreeBSD** για **embedded** εφαρμογές, κατ'αλληλες για χρήση από κάρτα Compact Flash (ή άλλο μέσο αποθήκευσης).

Θβιάέάò Ðåñéå÷:ïÿíúí

1 Εισαγωγή στο NanoBSD	1
2 Ρύθμιση του NanoBSD	2

1 Εισαγωγή στο **NanoBSD**

Το **NanoBSD** είναι ένα εργαλείο το οποίο αναπτύσσεται ενεργά από τον Poul-Henning Kamp. Δημιουργεί ένα εικονικό είδωλο του συστήματος **FreeBSD** για **embedded** εφαρμογές, ιδανικό για κάρτες Compact Flash (ή άλλο μέσο αποθήκευσης).

Μπορεί να χρησιμοποιηθεί για να φτιαχτούν εξειδικευμένες εικόνες, σχεδιασμένες για εύκολη εγκατάσταση και συντήρηση συστημάτων τύπου “computer appliance”. Τα συστήματα τέτοιου είδους έχουν το υλικό και λογισμικό τους ενσωματωμένα μέσα στο προϊόν. Αυτό σημαίνει πως συχνά είναι προεγκατεστημένα όλα τα απαραίτητα προγράμματα. Ε συσκευή συνδέεται σε ένα υπ'αρχον δίκτυο και μπορεί να λειτουργήσει (σχεδόν) αμέσως.

Τα βασικά χαρακτηριστικά του **NanoBSD** είναι:

- Τα ports και τα πακέτα λειτουργούν όπως στο FreeBSD — οποιοδήποτε πρόγραμμα μπορεί να εγκατασταθεί και να χρησιμοποιηθεί στο FreeBSD μπορεί να ενσωματωθεί και σε ένα είδωλο NanoBSD και να λειτουργήσει με τον ίδιο τρόπο.
- Δε υστερεί σε λειτουργικότητα — Αν είναι εφικτό να γίνει κάτι με το FreeBSD, τότε είναι εφικτό να γίνει το ίδιο πράγμα και με το NanoBSD, εκτός αν το συγκεκριμένο χαρακτηριστικό έχει σκόπιμα αφαιρεθεί από το είδωλο του NanoBSD.
- Το NanoBSD τρέχει με το σύστημα σε κατ'άσταση προστασίας από εγγραφή. Μπορείτε ακόμα και να το αποσυνδέσετε από το ρεύμα την ώρα που λειτουργεί. Δεν είναι αναγκαίο να εκτελεστεί το fsck(8) μετά από μια άγαρμψη διακοπή λειτουργίας του συστήματος.
- Είναι εύκολο να δημιουργηθεί και να προσαρμοστεί ένα είδωλο NanoBSD. Με τη χρήση ενός μόνο σεναρίου φλοιού και ενός αρχείου διαμόρφωσης είναι εφικτή η δόμηση προσαρμοσμένων ειδώλων που ικανοποιούν οποιαδήποτε ανάγκη.

2 Ρύθμιση του NanoBSD

2.1 Ε Σχεδίαση του NanoBSD

Αφού γραφτεί ένα είδωλο του NanoBSD σε κάποιο αποθηκευτικό μέσο, μπορεί να χρησιμοποιηθεί για να ξεκινήσει ένα σύστημα FreeBSD. Ε προκαθορισμένη διαμόρφωση του μέσου εκκίνησης αποτελείται από τρία μέρη:

- Δύο διαμερίσματα εκκίνησης: `code#1` και `code#2`.
- Το διαμέρισμα ρυθμίσεων, Αυτό μπορεί να προσαρτηθεί στον κατ'αλολο /`cfg` την ώρα λειτουργίας του NanoBSD.

Αυτά τα τρία διαμερίσματα κανονικά είναι προσαρτημένα μόνο για αν'αγνωση.

Οι κατ'αλολοι /`etc` και /`var` είναι εικονικοί δίσκοι md(4) (malloc).

Το διαμέρισμα ρυθμίσεων μπορεί να προσαρτηθεί στον κατ'αλολο /`cfg`. Περιέχει αρχεία τα οποία αντιγράφονται στον κατ'αλολο /`etc`. Κατ'α την εκκίνηση του συστήματος το διαμέρισμα ρυθμίσεων προσαρτ'αται προσωρινά μόνο για αν'αγνωση και αντιγράφονται τα αρχεία του στον κατ'αλολο /`etc`. Οπότε για να είναι κάποια αλλαγή ρυθμίσεων πιο "μόνιμη" και να ισχύει ακόμα και μετ'α από επανεκκίνηση του συστήματος, πρέπει να αντιγράφεται από τον κατ'αλολο /`etc` στο διαμέρισμα ρυθμίσεων, αφού αυτό προσαρτηθεί για γρ'αψιμο στον κατ'αλολο /`cfg`.

Διάγραμμα 1. Κ'ανοντας μόνιμες αλλαγές στο /`etc/resolv.conf`

```
# vi /etc/resolv.conf
[... ]
# mount /cfg
# cp /etc/resolv.conf /cfg
# umount /cfg
```

Όχι! Βύθος: Το διαμέρισμα το οποίο περιέχει τον κατ'αλολο /`cfg` πρέπει να προσαρτ'αται μόνο κατ'α την εκκίνηση και όταν έχουν προτεραιότητα τα αρχεία ρυθμίσεων του /`cfg`.

Δεν είναι καλή ιδέα να μένει μόνιμα προσαρτημένο το διαμέρισμα /cfg. Αν ένα διαμέρισμα είναι προσαρτημένο για γράψιμο και το αποθηκευτικό μέσο του **NanoBSD** επιτρέπει περιορισμένο αριθμό κύκλων εγγραφής (όπως π.χ. μερικές κάρτες CF), μπορεί να επηρεαστεί αρνητικά το αποθηκευτικό μέσο επειδή ο πυρήνας γράφει κάθε λίγο στους δίσκους του συστήματος (syncer).

2.2 Δημιουργία Ενός Προσαρμοσμένου Συστήματος NanoBSD

Το είδωλο του **NanoBSD** δημιουργείται από ένα απλό σενάριο κουνσόλας, το `nanobsd.sh`, το οποίο μπορεί να βρεθεί στον κατάλογο `/usr/src/tools/nanobsd`. Αυτό το σενάριο δημιουργεί ένα είδωλο δίσκου, το οποίο μπορεί να αντιγραφεί απευθείας σε κάποιο αποθηκευτικό μέσο χρησιμοποιώντας το πρόγραμμα `dd(1)`.

Οι απαραίτητες εντολές προκειμένου να δημιουργηθεί ένα είδωλο του **NanoBSD** είναι οι εξής:

```
# cd /usr/src/tools/tools/nanobsd ❶
# sh nanobsd.sh ❷
# cd /usr/obj/nanobsd.full ❸
# dd if=_.disk.full of=/dev/da0 bs=64k ❹
```

- ❶ Αλλαγή από τον τρέχοντα κατάλογο στον βασικό κατάλογο του σεναρίου δημιουργίας του **NanoBSD**.
- ❷ Εκκίνηση της δημιουργίας του ειδώλου.
- ❸ Αλλαγή από τον τρέχοντα κατάλογο στον κατάλογο που περιέχει το είδωλο του **NanoBSD**.
- ❹ Αντιγραφή του **NanoBSD** σε ένα μέσο αποθήκευσης.

2.3 Προσαρμόζοντας το είδωλο του NanoBSD

Αυτό είναι πιθανότατα το πιο σημαντικό και πιο ενδιαφέρον χαρακτηριστικό του **NanoBSD**. Αυτό είναι και το μέρος της διαδικασίας ανάπτυξης του ειδώλου **NanoBSD** για το οποίο θα ξοδέψετε ένα μεγάλο ποσοστό του χρόνου προετοιμασίας του ειδώλου.

Ε ακόλουθη εντολή θα κάνει το `nanobsd.sh` να διαβάσει τις ρυθμίσεις του από το αρχείο `myconf.nano` που βρίσκεται στο τρέχοντα κατάλογο:

```
# sh nanobsd.sh -c myconf.nano
```

Ε προσαρμογή και παραμετροποίηση του **NanoBSD** γίνεται συνήθως με δύο τρόπους:

- Επιλογές διαμόρφωσης
- Προσαρμοσμένες συναρτήσεις

2.3.1 Επιλογές διαμόρφωσης

Οι ρυθμίσεις του **NanoBSD** μπορούν να χρησιμοποιηθούν για να οριστούν προσαρμοσμένες επιλογές μεταγλώττισης και εγκατάστασης του **FreeBSD**. Αυτές οι επιλογές επηρεάζουν τα στάντα `buildworld` και `installworld` του **NanoBSD**, καθώς και άλλες επιλογές της δημιουργίας του τελικού

ειδώλου. Με τις κατ'αλληλες ρυθμίσεις το **NanoBSD** μπορεί να μικρύνει αρκετά σε μέγεθος· μπορεί να χωρέσει ένα πολύ μικρό υποσύστημα του βασικού FreeBSD σε χώρο μικρότερο από 64MB. Μπορείτε επίσης να χρησιμοποιήσετε τις επιλογές του **NanoBSD** για να δημιουργήσετε ένα είδωλο που έχει μόνο τον πυρήνα και δυο-τρία αρχεία στο βασικό σύστημα.

Το αρχείο ρυθμίσεων του **NanoBSD** περιέχει εντολές οι οποίες ορίζουν νέες τιμές για κάποια προκαθορισμένη μεταβλητή ή επεκτείνουν τις υπ'άρχουσες ρυθμίσεις. Οι πιο σημαντικές μεταβλητές είναι οι εξής:

- **NANO_NAME** — Το όνομα του ειδώλου **NanoBSD** που θα δημιουργηθεί (χρησιμοποιείται για να φτιάξει τα ονόματα των κατ'αλογων εργασιών).
- **NANO_SRC** — Ε διαδρομή του δέντρου πηγαιού κώδικα το οποίο θα χρησιμοποιηθεί για να μεταγλωττιστεί το είδωλο.
- **NANO_KERNEL** — Το όνομα του αρχείου ρυθμίσεων πυρήνα το οποίο θα χρησιμοποιηθεί για να χτιστεί ο πυρήνας του ειδώλου.
- **CONF_BUILD** — Επιλογές που περνούν στο στάδιο **buildworld** της μεταγλώττισης του FreeBSD.
- **CONF_INSTALL** — Επιλογές που περνούν στο στάδιο **installworld** της προετοιμασίας του ειδώλου.
- **CONF_WORLD** — Επιλογές που περνούν και στα δύο στάδια της προετοιμασίας του ειδώλου, και στο **buildworld** και στο **installworld**.
- **FlashDevice** — Ορίζει τον τύπο του μέσου αποθήκευσης που θα χρησιμοποιηθεί. Για περισσότερες λεπτομέρειες δείτε το αρχείο **FlashDevice.sub**.

2.3.2 Προσαρμοσμένες Συναρτήσεις

Στο αρχείο ρύθμισης του **NanoBSD** μπορεί να ρυθμιστεί σχεδόν κ'αθε λεπτομέρεια της προετοιμασίας του ειδώλου και του τελικού ειδώλου το οποίο θα δημιουργηθεί. Χρησιμοποιώντας μικρές συναρτήσεις φλοιού, μπορούμε να ρυθμίσουμε τα πάντα. Για παράδειγμα, μπορούμε να τρέξουμε δικές μας εντολές οι οποίες παραμετροποιούν το τελικό είδωλο ως εξής:

```
cust_foo () (
    echo "bar=topless" > \
        ${NANO_WORLDDIR}/etc/foo
)
customize_cmd cust_foo
```

Ένα πιο χρήσιμο παράδειγμα συνάρτησης φλοιού, η οποία ρυθμίζει κ'αποια παράμετρο του τελικού ειδώλου, είναι αυτό που ακολουθεί. Ε συνάρτηση **cust_etc_size** αλλάζει την προεπιλεγμένη τιμή μεγέθους του καταλόγου **/etc** από 5MB σε 30MB:

```
cust_etc_size () (
    cd ${NANO_WORLDDIR}/conf
    echo 30000 > default/etc/md_size
)
customize_cmd cust_etc_size
```

Το σενάριο φλοιού που δημιουργεί το **NanoBSD** έχει μερικές έτοιμες συναρτήσεις φλοιού οι οποίες εκτελούν τέτοιες επιπλέον ρυθμίσεις:

- `cust_comconsole` — Απενεργοποιεί το `getty(8)` στις κονσόλες VGA (οι συσκευές `/dev/ttyv*`) και ενεργοποιεί την χρήση της σειριακής θύρας COM1 ως κονσόλα συστήματος.
- `cust_allow_ssh_root` — Επιτρέπει την είσοδο του χρήστη `root` μέσω του `sshd(8)`.
- `cust_install_files` — Εγκαθιστά αρχεία από τον κατάλογο `nanobsd/Files`. Σε αυτό τον κατάλογο βρίσκονται μερικά χρήσιμα σενάρια φλοιού για τη διαχείριση ενός συστήματος **NanoBSD**.

2.3.3 Προσθέτοντας Πακέτα

Σε ένα είδωλο **NanoBSD** μπορούν να προστεθούν και έτοιμα πακέτα, που έχουν φτιαχτεί από πριν χρησιμοποιώντας τη Συλλογή των **Ports**. Μια συνάρτηση φλοιού μπορεί να εγκαταστήσει έξτρα πακέτα την ώρα που προετοιμάζεται το είδωλο. Για παράδειγμα, η παρακάτω συνάρτηση θα εγκαταστήσει στο είδωλο όλα τα πακέτα που βρίσκονται στον κατάλογο `/usr/src/tools/tools/nanobsd/packages`:

```
install_packages () (
mkdir -p ${NANO_WORLDDIR}/packages
cp /usr/src/tools/tools/nanobsd/packages/* ${NANO_WORLDDIR}/packages
chroot ${NANO_WORLDDIR} sh -c 'cd packages; pkg_add -v *;cd ..;'
rm -rf ${NANO_WORLDDIR}/packages
)

customize_cmd install_packages
```

2.3.4 Παράδειγμα Αρχείου Ρυθμίσεων του NanoBSD

Ένα πλήρες παράδειγμα αρχείου ρυθμίσεων για το **NanoBSD**, με διάφορες ρυθμίσεις για τη μεταγλώττιση του **FreeBSD** και την προετοιμασία του τελικού ειδώλου, μπορεί να μοιράζει κάπως έτσι:

```
NANO_NAME=custom
NANO_SRC=/usr/src
NANO_KERNEL=MYKERNEL
NANO_IMAGES=2

CONF_BUILD='
NO_KLDLOAD=YES
NO_NETGRAPH=YES
NO_PAM=YES
'

CONF_INSTALL='
NO_ACPI=YES
NO_BLUETOOTH=YES
NO_CVS=YES
NO_FORTRAN=YES
NO_HTML=YES
NO_LPR=YES
NO_MAN=YES
NO_SENDMAIL=YES
```

```
NO_SHAREDOKS=YES
NO_EXAMPLES=YES
NO_INSTALLLIB=YES
NO_CALENDAR=YES
NO_MISC=YES
NO_SHARE=YES
'
```

```
CONF_WORLD='
NO_BIND=YES
NO_MODULES=YES
NO_KERBEROS=YES
NO_GAMES=YES
NO_RESCUE=YES
NO_LOCALES=YES
NO_SYSCONS=YES
NO_INFO=YES
'
```

```
FlashDevice SanDisk 1G
```

```
cust_nobeastie() (
    touch ${NANO_WORLDDIR}/boot/loader.conf
    echo "beastie_disable=\"YES\" >> ${NANO_WORLDDIR}/boot/loader.conf
)
```

```
customize_cmd cust_comconsole
customize_cmd cust_install_files
customize_cmd cust_allow_ssh_root
customize_cmd cust_nobeastie
```

2.4 Ενημερώνοντας το NanoBSD

Η διαδικασία ενημέρωσης του **NanoBSD** είναι σχετικά απλή:

1. Δημιουργία ενός νέου ειδώλου **NanoBSD**, με το συνηθισμένο τρόπο.
2. Ανέβασμα του νέου ειδώλου σε μια χρησιμοποιήσιμη κατ'ατμηση μιας συσκευής που τρέχει **NanoBSD**.

Ε πιο σημαντική διαφορά αυτού του σημείου από την αρχική εγκατάσταση του **NanoBSD** είναι ότι τώρα αντί του αρχείου `_.disk.full` (το οποίο περιέχει το είδωλο ενός ολόκληρου δίσκου), εγκαταστάθηκε το αρχείο `_.disk.image` (το οποίο περιέχει το είδωλο μιας μόνο κατ'ατμησης του δίσκου).

3. Κλείσιμο και επανεκκίνηση του συστήματος από την νέα εγκατεστημένη κατ'ατμηση.
4. Αν όλα π'ανε καλ'α, η αναβάθμιση τελείωσε.
5. Αν οτιδήποτε π'αιε στραβ'α, επανεκκινήστε τη συσκευή από την προηγούμενη κατ'ατμηση (η οποία περιέχει το παλιό είδωλο, που λειτουργεί σωστ'α). Έτσι μπορείτε να επαναφέρετε άμεσα το

σύστημα σε λειτουργική κατάσταση. Διορθώστε οποιαδήποτε προβλήματα έχει το νέο είδωλο, και επαναλάβετε την διαδικασία.

Για να εγκατασταθεί το νέο είδωλο σε ένα σύστημα **NanoBSD**, το οποίο ήδη βρίσκεται σε λειτουργία, μπορεί να χρησιμοποιηθεί είτε το σενάριο **updatep1** είτε το **updatep2**. Αυτά τα δύο σενάρια φλοιού βρίσκονται στον κατάλογο **/root**.

Ανάλογα με τις υπηρεσίες του συστήματος στο οποίο δημιουργείται ένα είδωλο **NanoBSD**, μπορεί να μεταφερθεί ένα νέο είδωλο στο τελικό σύστημα με διάφορους τρόπους:

2.4.1 Χρησιμοποιώντας το **ftp(1)**

Αν σας ενδιαφέρει η καλή ταχύτητα μεταφοράς, χρησιμοποιήστε το FTP για τη μεταφορά του ειδώλου:

```
# ftp myhost
get _.disk.image "| sh updatep1"
```

2.4.2 Χρησιμοποιώντας το **ssh(1)**

Αν σας ενδιαφέρει η μεταφορά του ειδώλου να γίνει με ασφαλή τρόπο και δεν έχετε αφαιρέσει από το αρχικό είδωλο το **ssh(1)**, προτιμήστε τη μεταφορά μέσω SSH:

```
# ssh myhost cat _.disk.image.gz | zcat | sh updatep1
```

2.4.3 Χρησιμοποιώντας το **nc(1)**

Αν το σύστημα στο οποίο έχει δημιουργηθεί το νέο είδωλο του **NanoBSD** δεν τρέχει ούτε **ftpd(8)** ούτε **sshd(8)**, μπορείτε να χρησιμοποιήσετε άλλα εργαλεία για τη μεταφορά, όπως το **nc(1)**:

1. Πρώτα ανοίξτε ένα εξυπηρετητή TCP στο σύστημα το οποίο θα παρέχει το νέο είδωλο, και ορίστε το νέο είδωλο ως είσοδο για τη μεταφορά που θα γίνει αργότερα:

```
myhost# nc -l 2222 < _.disk.image
```

Όχι!Βυός: Πριν από αυτό βεβαιωθείτε ότι η θύρα που χρησιμοποιείτε για τον εξυπηρετητή μπορεί να δεχθεί εισερχόμενες συνδέσεις από το σύστημα **NanoBSD** προς το σύστημα του εξυπηρετητή, και δεν εμποδίζεται, για παράδειγμα, από κάποιο τείχος προστασίας ("firewall").

2. Συνδεθείτε από το **NanoBSD** στον εξυπηρετητή, ο οποίος είναι ήδη έτοιμος να παρέχει το νέο είδωλο, και τροφοδοτήστε το σενάριο φλοιού **updatep1** με την έξοδο του εργαλείου **nc(1)**:

```
# nc myhost 2222 | sh updatep1
```