Συχνές Ερωτήσεις για το FreeBSD 12.X και 13.X

Περίληψη

Το κείμενο αυτό περιέχει τις Συχνές Ερωτήσεις για τις εκδόσεις 6.*X* και 7.*X* του FreeBSD. Όλες οι καταχωρήσεις θεωρείται ότι ισχύουν για την έκδοση 6.*X* και τις μεταγενέστερες εκδόσεις, εκτός αν αναφέρεται διαφορετικά. Αν ενδιαφέρεστε να μας βοηθήσετε σε αυτό το έργο, στείλτε ένα email στην ηλεκτρονική λίστα ομάδας τεκμηρίωσης του FreeBSD. Η τελευταία έκδοση αυτού του κειμένου είναι πάντοτε διαθέσιμη στην δικτυακή τοποθεσία του FreeBSD. Μπορείτε επίσης να την κατεβάσετε ως ένα μεγάλο αρχείο HTML μέσω HTTP ή ακόμα και ως απλό κείμενο, PostScript®, PDF, κλπ. από τον εξυπηρετητή FTP του FreeBSD. Μπορείτε ακόμα να ψάξετε στις Συχνές Ερωτήσεις.

Πίνακας Περιεχομένων

1. Εισαγωγή . . . 3
2. Τεκμηρίωση και Υποστήριξη . . . 9
3. Εγκατάσταση . . . 14
4. Συμβατότητα Υλικού . . . 26
5. Αντιμετώπιση Προβλημάτων . . . 36
6. Εμπορικές Εφαρμογές . . . 54
7. Εφαρμογές Τελικού Χρήστη . . . 57
8. Ρύθμιση Πυρήνα . . . 61
9. Δίσκοι, Συστήματα Αρχείων και Φορτωτές Εκκίνησης . . . 64
10. Διαχείριση Συστήματος . . . 77
11. Το Σύστημα X Windows και οι Εικονικές Κονσόλες . . . 87
12. Δικτύωση . . . 98
13. Ασφάλεια . . . 106
14. PPP . . . 110
15. Σειριακές Επικοινωνίες . . . 124
16. Διάφορες Ερωτήσεις . . . 132
17. Χιούμορ και FreeBSD . . . 137
18. Προχωρημένα Θέματα . . . 140
19. Ευχαριστίες . . . 148
Bibliography . . . 149

Chapter 1. Εισαγωγή

Καλώς ήλθατε στις Συχνές Ερωτήσεις του FreeBSD 6.*X*-7.*X*!

Όπως συνηθίζεται στα FAQs του Usenet, το κείμενο αυτό έχει σκοπό να καλύψει τις πιο συνηθισμένες ερωτήσεις που αφορούν το λειτουργικό σύστημα FreeBSD (και φυσικά να τις απαντήσει!). Αν και ο αρχικός σκοπός των FAQs ήταν η εξοικονόμηση εύρους ζώνης του διαδικτύου από την επαναλαμβανόμενη απάντηση των ίδιων ερωτήσεων, τα FAQs αναγνωρίστηκαν τελικά ως πολύτιμες πηγές πληροφοριών.

Έχει καταβληθεί κάθε προσπάθεια ώστε αυτό το FAQ να περιέχει όσο το δυνατόν περισσότερες πληροφορίες. Αν θέλετε να κάνετε κάποιες υποδείξεις σχετικά με τρόπους βελτίωσης του, σας παρακαλούμε στείλτε μας ένα email στη ηλεκτρονική λίστα ομάδας τεκμηρίωσης του FreeBSD.

1.1. Τι είναι το FreeBSD;

Εν συντομία, το FreeBSD είναι ένα λειτουργικό σύστημα τύπου UNIX® για τις αρχιτεκτονικές AMD64 και Intel® EM64T, i386™ PC-98, IA-64, ARM®, PowerPC® και UltraSPARC®. Βασίζεται στην έκδοση "4.4BSD-Lite" του Πανεπιστημίου του Berkeley, με κάποιες βελτιώσεις από το "4.4BSD-Lite2". Βασίζεται επίσης έμμεσα στην εργασία μεταφοράς (port) του "Net/2" (επίσης από το Berkeley) που έγινε από τον William Jolitz (η οποία έγινε γνωστή με το όνομα "386BSD"), αν και δεν έχει μείνει σχεδόν τίποτα από τον αρχικό κώδικα του 386BSD. Πληρέστερη περιγραφή για το τι είναι το FreeBSD και πως μπορεί να λειτουργήσει για εσάς, μπορείτε να βρείτε στην δικτυακή τοποθεσία του FreeBSD.

Το FreeBSD χρησιμοποιείται από εταιρίες, παροχείς υπηρεσιών Internet (ISPs), ερευνητές, επαγγελματίες της Πληροφορικής, φοιτητές και οικιακούς χρήστες από όλο τον κόσμο, στη δουλειά τους, στην εκπαίδευση και για αναψυχή.

Για περισσότερες λεπτομέρειες σχετικά με το FreeBSD, παρακαλούμε διαβάστε το Εγχειρίδιο του FreeBSD.

1.2. Ποιος είναι ο σκοπός του FreeBSD Project;

Ο σκοπός του FreeBSD Project είναι να παρέχει λογισμικό το οποίο να μπορεί να χρησιμοποιηθεί για οποιουσδήποτε σκοπούς και χωρίς περιορισμούς. Πολλοί από εμάς έχουμε επενδύσει σημαντικά στον κώδικα (και στο ίδιο το έργο) και σίγουρα δεν θα μας πείραζε και κάποια χρηματική αποζημίωση που και που, αλλά σίγουρα δεν επιμένουμε σε αυτό. Πιστεύουμε ότι η πρώτη και βασικότερη "αποστολή" μας είναι να παρέχουμε κώδικα σε όλους, για οποιοδήποτε πιθανό σκοπό, έτσι ώστε ο κώδικας μας να έχει την ευρύτερη δυνατή χρήση και επιτυγχάνει το μεγαλύτερο δυνατό όφελος. Πιστεύουμε ότι αυτός είναι ένας από τους σημαντικότερους σκοπούς του Ελεύθερου Λογισμικού και τον υποστηρίζουμε θερμά.

Ο κώδικας, στο δέντρο του πηγαίου μας κώδικα, ο οποίος εμπίπτει στη άδεια χρήσης GNU General Public License (GPL) ή GNU Library General Public License (LGPL) έχει περισσότερους περιορισμούς, αλλά αυτοί κλίνουν, ευτυχώς, προς την υποχρέωση ελεύθερης πρόσβασης αντί για το αντίθετο που είναι και το πλέον συνηθισμένο. Λόγω της αυξημένης πολυπλοκότητας που μπορεί να προκύψει από την εμπορική χρήση λογισμικού GPL, προσπαθούμε γενικά να αντικαταστήσουμε

αυτό το λογισμικό με αντίστοιχο υπό την πιο χαλαρή άδεια FreeBSD, όπου αυτό είναι δυνατό.

1.3. Υπάρχουν κάποιοι περιορισμοί στην άδεια του FreeBSD;

Ναι. Οι περιορισμοί αυτοί δεν ελέγχουν πως χρησιμοποιείτε τον κώδικα, αλλά μόνο πως μεταχειρίζεστε το ίδιο το FreeBSD Project. Αν σας ενδιαφέρει σοβαρά η άδεια, διαβάστε την εδώ. Για τους απλώς περίεργους, η άδεια περιληπτικά γράφει:

- Μην ισχυρισθείτε ότι γράψατε αυτό το λογισμικό.
- Μη μας μηνύσετε αν χαλάσει.

1.4. Μπορεί το FreeBSD να αντικαταστήσει το τρέχον λειτουργικό μου σύστημα;

Για τους περισσότερους ανθρώπους, ναι. Αλλά αυτή η απάντηση δεν μπορεί να δοθεί τόσο ξερά.

Οι περισσότεροι άνθρωποι δεν χρησιμοποιούν στην πραγματικότητα κάποιο λειτουργικό σύστημα. Χρησιμοποιούν εφαρμογές. Οι εφαρμογές είναι αυτές που χρησιμοποιούν το λειτουργικό σύστημα. Το FreeBSD είναι σχεδιασμένο για να παρέχει ένα στιβαρό και ολοκληρωμένο περιβάλλον για εφαρμογές. Υποστηρίζει μεγάλη ποικιλία από φυλλομετρητές, σουίτες γραφείου, προγράμματα ηλεκτρονικού ταχυδρομείου, προγράμματα γραφικών, περιβάλλοντα προγραμματισμού, εξυπηρετητές δικτύου, και ουσιαστικά οτιδήποτε άλλο μπορεί να θελήσετε. Μπορείτε να διαχειριστείτε τις περισσότερες από αυτές τις εφαρμογές μέσω της Συλλογής των Ports.

Αν χρειάζεται να χρησιμοποιήσετε μια εφαρμογή που είναι διαθέσιμη σε ένα μόνο λειτουργικό σύστημα, τότε απλά δεν μπορείτε να αντικαταστήσετε αυτό το λειτουργικό σύστημα. Είναι, όμως, πολύ πιθανό ότι θα βρείτε μια αντίστοιχη εφαρμογή στο FreeBSD. Αν θέλετε έναν σταθερό εξυπηρετητή για το γραφείο σας ή για το Internet, ένα αξιόπιστο σταθμό εργασίας, ή απλώς την ικανότητα να κάνετε την εργασία σας χωρίς διακοπές, είναι σχεδόν σίγουρο ότι το FreeBSD μπορεί να κάνει ότι χρειάζεστε. Πολλοί χρήστες υπολογιστών σε όλο τον κόσμο - τόσο αρχάριοι όσο και έμπειροι διαχειριστές συστημάτων UNIX® - χρησιμοποιούν το FreeBSD ως το βασικό λειτουργικό σύστημα στους σταθμούς εργασίας τους.

Αν έρχεστε στο FreeBSD από κάποιο άλλο περιβάλλον τύπου UNIX®, ξέρετε ήδη τα περισσότερα από αυτά που χρειάζεστε. Αν ωστόσο η εμπειρία σας προέρχεται από λειτουργικά συστήματα με γραφικό περιβάλλον, όπως τα Windows® και οι παλιότερες εκδόσεις του Mac OS®, θα πρέπει να επενδύσετε, όπως είναι αναμενόμενο, επιπλέον χρόνο για να εξοικειωθείτε με τον τρόπο εργασίας σε ένα σύστημα UNIX®. Αυτό το FAQ, καθώς και το Εγχειρίδιο του FreeBSD αποτελούν εξαιρετικά αναγνώσματα για να ξεκινήσετε.

1.5. Γιατί ονομάζεται FreeBSD;

- Μπορεί να χρησιμοποιηθεί χωρίς χρέωση, ακόμα και για εμπορικούς σκοπούς.
- Διατίθεται ο πλήρης πηγαίος κώδικας για το λειτουργικό σύστημα, και με τους ελάχιστους δυνατούς περιορισμούς σχετικά με τη χρήση του, τη διανομή του και την ενσωμάτωση του σε

άλλα έργα (εμπορικά ή μη).

- Οποιοσδήποτε έχει να προτείνει μια βελτίωση ή διόρθωση, είναι ελεύθερος να υποβάλλει τον κώδικα του, ο οποίος και θα προστεθεί στο δέντρο πηγαίου κώδικα (αρκεί να πληρούνται μια-δυο βασικές προφανείς προϋποθέσεις).

Αξίζει να αναφέρουμε ότι η λέξη "ελεύθερο" χρησιμοποιείται εδώ με δύο τρόπους, ο ένας σημαίνει "δωρεάν", και ο άλλος σημαίνει "μπορείτε να το κάνετε ότι θέλετε". Εκτός από ένα-δύο πράγματα που *δεν* μπορείτε να κάνετε με τον κώδικα του FreeBSD, για παράδειγμα να ισχυρισθείτε ότι τον γράψατε, μπορείτε στα αλήθεια να τον κάνετε ότι θέλετε.

1.6. Ποιες είναι οι διαφορές μεταξύ του FreeBSD και του NetBSD, OpenBSD και των άλλων BSD συστημάτων ανοικτού κώδικα;

Ο James Howard έχει γράψει μια καλή εξήγηση της ιστορίας και των διαφορών μεταξύ των διάφορων έργων για το DaemonNews, στο άρθρο Η Οικογένεια Λειτουργικών BSD.

1.7. Ποια είναι η τελευταία έκδοση του FreeBSD;

Αυτή τη στιγμή υπάρχουν δύο παράλληλοι κλάδοι στην ανάπτυξη του FreeBSD. Οι επίσημες εκδόσεις του FreeBSD παράγονται ταυτόχρονα και από τους δύο αυτούς κλάδους. Η σειρά των εκδόσεων 6.*X* δημιουργείται από τον κλάδο *6-STABLE* και η σειρά εκδόσεων 7.*X* δημιουργείται από τον κλάδο *7-STABLE*.

Μέχρι και την έκδοση 7.0, η σειρά εκδόσεων 6.*X* ήταν γνωστή ως *-STABLE*. Ωστόσο από την 7.0 και μετά, η σειρά 6.*X* θεωρείται ότι εισάγεται στη φάση της "εκτεταμένης υποστήριξης" και θα λαμβάνει μόνο διορθώσεις για σοβαρά προβλήματα, όπως π.χ. αυτά που αναφέρονται σε κενά ασφαλείας. Θα υπάρξουν περισσότερες εκδόσεις από τη σειρά *6-STABLE*, αλλά αυτή θεωρείται πλέον "ξεπερασμένη" (legacy) και οι πιο πρόσφατες βελτιώσεις θα ενσωματωθούν μόνο στον κλάδο *7-STABLE*.

Version 12.0 is the latest release from the *7-STABLE* branch; it was released in December 11, 2018. Version 11.2 is the latest release from the *6-STABLE* branch; it was released in June 28, 2018.

Η έκδοση 12.0 είναι η πλέον τελευταία της σειράς *7-STABLE* και δημιουργήθηκε στις December 11, 2018. Η έκδοση 11.2 είναι η πλέον τελευταία της σειράς *6-STABLE* και δημιουργήθηκε στις June 28, 2018.

Εν συντομία, η σειρά *-STABLE* προορίζεται για τους ISPs, τους εταιρικούς χρήστες καθώς και οποιοδήποτε χρήστη θέλει σταθερότητα και μικρό αριθμό αλλαγών σε σχέση με τα νέα (και πιθανόν ασταθή) χαρακτηριστικά που εμφανίζονται στη σειρά *-CURRENT*. Εκδόσεις μπορεί να γίνονται από οποιοδήποτε κλάδο, αλλά η έκδοση *-CURRENT* θα πρέπει να χρησιμοποιείται μόνο αν είστε προετοιμασμένοι να χειριστείτε τα ενδεχομένως πιο ασταθή χαρακτηριστικά της (σε σχέση πάντα με την αντίστοιχη *-STABLE*).

Νέες εκδόσεις εμφανίζονται κάθε μερικούς μήνες. Αν και πολλοί επιλέγουν να διατηρούνται ενημερωμένοι και πιο συχνά μέσω του πηγαίου κώδικα του FreeBSD (δείτε τις ερωτήσεις στο

FreeBSD-CURRENT και FreeBSD-STABLE), οι εκδόσεις είναι κάτι παραπάνω από υποχρέωση, καθώς ο πηγαίος κώδικας είναι περισσότερο ένας κινούμενος στόχος.

Περισσότερες πληροφορίες για τις εκδόσεις του FreeBSD μπορείτε να βρείτε στη Σελίδα Προετοιμασίας Εκδόσεων στην δικτυακή τοποθεσία του FreeBSD.

1.8. Τι είναι το FreeBSD-CURRENT;

Το FreeBSD-CURRENT είναι η υπό εξέλιξη έκδοση του λειτουργικού συστήματος, το οποίο με τον καιρό θα εξελιχθεί στο νέο κλάδο FreeBSD-STABLE. Για το λόγο αυτό, παρουσιάζει συνήθως ενδιαφέρον μόνο σε όσους ασχολούνται με την ανάπτυξη κώδικα του συστήματος και σε σκληροπυρηνικούς χομπίστες. Δείτε το σχετικό τμήμα στο εγχειρίδιο για λεπτομέρειες σχετικά με το -CURRENT.

Αν δεν είστε εξοικειωμένος με το λειτουργικό σύστημα, ή δεν είστε ικανός να αναγνωρίσετε τη διαφορά μεταξύ ενός πραγματικού και ενός προσωρινού προβλήματος, μάλλον δεν θα πρέπει να χρησιμοποιήσετε το FreeBSD-CURRENT. Ο κλάδος αυτός ορισμένες φορές εξελίσσεται πολύ γρήγορα, και μπορεί ακόμα και η μεταγλώττιση του να μην είναι δυνατή για αρκετές συνεχόμενες μέρες. Όσοι χρησιμοποιούν το FreeBSD-CURRENT αναμένεται να είναι ικανοί να αναλύουν τα όποια προβλήματα και να τα αναφέρουν μόνο αν θεωρούν ότι πρόκειται για σημαντικά λάθη και όχι για "μικροπροβλήματα". Ερωτήσεις του τύπου "το make world παράγει κάποια σφάλματα σχετικά με τα groups" στη λίστα ταχυδρομείου -CURRENT, μπορεί να αντιμετωπιστούν περιφρονητικά.

Κάθε μήνα, παράγονται στιγμιότυπα εκδόσεων που βασίζονται στην τρέχουσα κατάσταση των κλάδων *-CURRENT* και *-STABLE*. Οι στόχοι πίσω από κάθε έκδοση στιγμιότυπου είναι:

- Ο έλεγχος της τελευταίας έκδοσης του λογισμικού εγκατάστασης.
- Να δώσει τη δυνατότητα εύκολης εγκατάστασης σε όσους επιθυμούν να εγκαταστήσουν το *-CURRENT* ή το *-STABLE* αλλά δεν έχουν το χρόνο ή το εύρος ζώνης να το παρακολουθούν μέρα με τη μέρα.
- Η διατήρηση ενός σταθερού σημείου αναφοράς σχετικά με τον πηγαίο κώδικα, σε περίπτωση που χαλάσουμε κάτι πολύ άσχημα αργότερα. (Αν και λόγω της χρήσης του CVS είναι δύσκολο να συμβεί κάτι πραγματικά τόσο φρικτό.)
- Να εξασφαλιστεί ότι κάθε νέο χαρακτηριστικό και διόρθωση που χρειάζεται έλεγχο, θα έχει το μεγαλύτερο δυνατό κοινό πιθανών δοκιμαστών.

Δεν παρέχεται καμία εγγύηση "τελικής ποιότητας" για τα στιγμιότυπα του κλάδου *-CURRENT*. Αν χρειάζεστε ένα σταθερό και δοκιμασμένο σύστημα, θα πρέπει να προτιμάτε τις επίσημες πλήρεις εκδόσεις ή τα στιγμιότυπα του κλάδου *-STABLE*.

Περισσότερες πληροφορίες για τα στιγμιότυπα μπορείτε να βρείτε στη σελίδα στιγμιότυπων εκδόσεων.

Επίσημα στιγμιότυπα εκδόσεων παράγονται μια φορά το μήνα για όλους τους ενεργούς κλάδους ανάπτυξης. Ημερήσια στιγμιότυπα για τις πιο δημοφιλείς αρχιτεκτονικές (i386 και amd64) μπορείτε να βρείτε στη σελίδα http://snapshots.us.freebsd.org/.

1.9. Τι είναι το FreeBSD-STABLE;

Όταν κυκλοφόρησε το FreeBSD 2.0.5, η ανάπτυξη του FreeBSD χωρίστηκε σε δύο κλάδους. Ο ένας κλάδος ονομάστηκε -STABLE, και ο άλλος -CURRENT. Το *FreeBSD-STABLE* απευθύνεται σε Παροχείς Υπηρεσιών Internet (ISPs) και άλλες εμπορικές χρήσεις, όπου οι απότομες αλλαγές και τα τυχόν πειραματικά χαρακτηριστικά είναι γενικά ανεπιθύμητα. Στον κλάδο αυτό ενσωματώνονται μόνο καλά δοκιμασμένες διορθώσεις και άλλες μικρές αλλαγές. Από την άλλη μεριά, το *FreeBSD-CURRENT* βρίσκεται σε μια μοναδική αδιάκοπη γραμμή ανάπτυξης από την εποχή της κυκλοφορίας της έκδοσης 2.0, οδηγώντας προς την έκδοση 12.0-RELEASE αλλά και μετά από αυτή. Για περισσότερες πληροφορίες δείτε το άρθρο "Προετοιμασία Εκδόσεων του FreeBSD: Δημιουργία του Κλάδου Έκδοσης".

Ο κλάδος 2.2-STABLE εγκαταλείφθηκε με την κυκλοφορία της έκδοσης 2.2.8. Ο κλάδος 3-STABLE τελείωσε με την κυκλοφορία της έκδοσης 3.5.1, που ήταν και η τελευταία της σειράς εκδόσεων 3.*X*. Ο κλάδος 4-STABLE τελείωσε με την κυκλοφορία της έκδοσης 4.11, της τελευταίας της σειράς 4.*X*. Οι μόνες αλλαγές που γίνονται κατά βάση σε κάθε ένα από αυτούς τους κλάδους, έχουν σχέση με διορθώσεις σε κενά ασφαλείας. Η υποστήριξη των κλάδων 5-STABLE σταμάτησε με την έκδοση 5.5, την τελευταία έκδοση της σειράς 5.*X*. Η υποστήριξη για τη σειρά εκδόσεων 6-STABLE θα συνεχιστεί για λίγο καιρό ακόμα, αλλά θα επικεντρωθεί περισσότερο σε διορθώσεις κενών ασφαλείας και άλλων σοβαρών προβλημάτων.

Ο κλάδος 12.0-STABLE είναι ο -*STABLE* κλάδος που βρίσκεται υπό ενεργή ανάπτυξη. Η τελευταία κυκλοφορία που βασίζεται στον κλάδο 12.0-STABLE είναι η 12.0-RELEASE, και έγινε στις December 11, 2018.

Ο κλάδος 8-CURRENT είναι ο κλάδος -*CURRENT* που αναπτύσσεται αυτή τη στιγμή ενεργά ώστε να δημιουργηθεί η νέα γενιά του FreeBSD. Δείτε το τμήμα Τι είναι το FreeBSD-CURRENT; για περισσότερες πληροφορίες σχετικά με αυτό τον κλάδο.

1.10. Κάθε πότε γίνονται επίσημες κυκλοφορίες του FreeBSD;

Η Ομάδα Οργάνωσης των Εκδόσεων <re@FreeBSD.org> δίνει στην κυκλοφορία μια καινούρια έκδοση του FreeBSD κάθε περίπου τέσσερις μήνες, κατά μέσο όρο. Οι ημερομηνίες κυκλοφορίας ανακοινώνονται αρκετό καιρό πριν, ώστε όσοι δουλεύουν πάνω στο σύστημα να ξέρουν πότε οι εργασίες τους πρέπει να έχουν ολοκληρωθεί και δοκιμαστεί. Πριν από κάθε κυκλοφορία, προηγείται μια περίοδος δοκιμών, ώστε να εξασφαλιστεί ότι η προσθήκη νέων χαρακτηριστικών δεν έχει αρνητικές επιπτώσεις στη σταθερότητα της έκδοσης. Πολλοί χρήστες θεωρούν αυτό το είδος της προσοχής ένα από τα καλύτερα πράγματα σχετικά με το FreeBSD, αν και η αναμονή ως ότου φτάσουν όλα τα ωραία νέα χαρακτηριστικά το -STABLE μπορεί να γίνει λίγο εκνευριστική.

Περισσότερες πληροφορίες σχετικά με τη διαδικασία κυκλοφορίας (περιλαμβανομένου και ενός χρονοδιαγράμματος επικείμενων κυκλοφοριών) μπορούν να βρεθούν στις σελίδες προετοιμασίας εκδόσεων στη δικτυακή τοποθεσία του FreeBSD.

Για τους πιο ενθουσιώδεις, υπάρχουν καθημερινά στιγμιότυπα (binary snapshots) όπως αναφέρθηκε πιο πάνω.

1.11. Ποιος είναι υπεύθυνος για το FreeBSD

Οι βασικές αποφάσεις που αφορούν το FreeBSD Project, όπως η συνολική κατεύθυνση του έργου και ποιος επιτρέπεται να προσθέσει πηγαίο κώδικα στο δέντρο, παίρνονται από την βασική ομάδα (core team) που αποτελείται από 9 άτομα. Υπάρχει μια ακόμα μεγαλύτερη ομάδα με περισσότερα από 350 άτομα που ονομάζονται διαπράκτες (committers) και οι οποίοι έχουν την έγκριση να κάνουν απευθείας αλλαγές στο δέντρο πηγαίου κώδικα του FreeBSD.

Ωστόσο οι περισσότερες σημαντικές αλλαγές συζητούνται από πριν στις λίστες ηλεκτρονικού ταχυδρομείου, και δεν υπάρχουν περιορισμοί σχετικά με το ποιοι λαμβάνουν μέρος σε αυτές τις συζητήσεις.

1.12. Που μπορώ να προμηθευτώ το FreeBSD;

Κάθε σημαντική έκδοση του FreeBSD είναι διαθέσιμη μέσω ανώνυμου FTP από τον διακομιστή FTP του FreeBSD:

- Η τελευταία κυκλοφορία που βασίζεται στο 6-STABLE, 12.0-RELEASE μπορεί να βρεθεί στον κατάλογο 12.0-RELEASE.
- Εκδόσεις που βασίζονται σεΣτιγμιότυπα γίνονται καθημερινά από τους κλάδους -CURRENT και -STABLE, και εξυπηρετούν κατά βάση όσους ασχολούνται με την ανάπτυξη και τον έλεγχο των τελευταίας γενιάς προγραμμάτων.
- Η τελευταία κυκλοφορία που βασίζεται στον κλάδο 5-STABLE, η 11.2-RELEASE, μπορεί να βρεθεί στον κατάλογο 11.2-RELEASE.

Πληροφορίες σχετικά με τη διάθεση του FreeBSD σε CD, DVD και άλλα μέσα μπορούν να βρεθούν στο Εγχειρίδιο.

1.13. Πως μπορώ να έχω πρόσβαση στη Βάση Δεδομένων με τις Αναφορές Προβλημάτων;

Μπορείτε να κάνετε αναζητήσεις στη βάση δεδομένων με τις αναφορές προβλημάτων μέσω της διεπαφής αναζητήσεων στο Web.

Η εντολή send-pr(1) μπορεί να χρησιμοποιηθεί για την υποβολή αναφορών προβλημάτων, καθώς και αιτήσεων για αλλαγές, μέσω ηλεκτρονικού ταχυδρομείου. Εναλλακτικά, μπορείτε να χρησιμοποιήσετε την αντίστοιχη δυνατότητα υποβολής αναφορών προβλημάτων μέσω της διεπαφής web με τη βοήθεια ενός προγράμματος φυλλομετρητή.

Πριν υποβάλετε μια αναφορά προβλήματος, παρακαλούμε διαβάστε το Γράφοντας Αναφορές Προβλημάτων για το FreeBSD, ένα άρθρο για το πως να γράφετε καλές αναφορές προβλημάτων.

1.14. Τι άλλες πηγές πληροφοριών υπάρχουν;

Παρακαλούμε ελέγξτε τη λίστα της Τεκμηρίωσης στην κύρια δικτυακή τοποθεσία του FreeBSD.

Chapter 2. Τεκμηρίωση και Υποστήριξη

2.1. Ποια καλά βιβλία υπάρχουν σχετικά με το FreeBSD;

Το Project παράγει μια μεγάλη γκάμα τεκμηρίωσης που διατίθεται online από τον παρακάτω σύνδεσμο: http://www.FreeBSD.org/docs/. Τα ίδια αυτά έγγραφα είναι διαθέσιμα και ως πακέτα τα οποία μπορείτε να εγκαταστήσετε εύκολα στο FreeBSD σύστημα σας. Περισσότερες λεπτομέρειες σχετικά με τα πακέτα τεκμηρίωσης, θα βρείτε στις ακόλουθες παραγράφους.

Επιπρόσθετα, θα βρείτε και άλλα συνιστώμενα βιβλία στην Βιβλιογραφία στο τέλος αυτού του FAQ, και του Εγχειριδίου.

2.2. Είναι διαθέσιμη η τεκμηρίωση και σε άλλες μορφές, όπως απλό κείμενο (ASCII) ή PostScript®;

Ναι. Η τεκμηρίωση είναι διαθέσιμη σε πλήθος διαφορετικών μορφών και τρόπων συμπίεσης, στο διακομιστή FTP του FreeBSD, στον κατάλογο /pub/FreeBSD/doc/.

Η τεκμηρίωση είναι κατηγοριοποιημένη με διάφορους τρόπους. Αυτοί περιλαμβάνουν:

- Το όνομα του εγγράφου, όπως π.χ. `faq`, ή `handbook`.
- Η γλώσσα και η κωδικοποίηση του κειμένου. Αυτά βασίζονται στα ονόματα που έχουν δοθεί στις τοπικές ρυθμίσεις και που μπορείτε να βρείτε στον κατάλογο /usr/shared/locale στο FreeBSD σύστημά σας. Οι τρέχουσες γλώσσες και κωδικοποιήσεις που έχουμε διαθέσιμες αυτή τη στιγμή στην τεκμηρίωση είναι οι παρακάτω:

Όνομα	Ερμηνεία
`en_US.ISO8859-1`	Αγγλικά ΗΠΑ
`bn_BD.ISO10646-1`	Bengali (ή Bangla)
`da_DK.ISO8859-1`	Δανέζικα
`de_DE.ISO8859-1`	Γερμανικά
`es_ES.ISO8859-1`	Ισπανικά
`fr_FR.ISO8859-1`	Γαλλικά
`hu_HU.ISO8859-2`	Ουγγαρέζικα
`it_IT.ISO8859-15`	Ιταλικά
`ja_JP.eucJP`	Γιαπωνέζικα (κωδικοποίηση EUC)
`mn_MN.UTF-8`	Μογγολικά (κωδικοποίηση UTF-8)
`nl_NL.ISO8859-1`	Ολλανδικά
`pl_PL.ISO8859-2`	Πολωνικά

Όνομα	Ερμηνεία
`pt_BR.ISO8859-1`	Πορτογαλικά (Βραζιλία)
`ru_RU.KOI8-R`	Ρώσικα (κωδικοποίηση KOI8-R)
`sr_YU.ISO8859-2`	Σέρβικα
`tr_TR.ISO8859-9`	Τούρκικα
`zh_CN.GB2312`	Απλοποιημένα Κινέζικα (κωδικοποίηση GB2312)
`zh_TW.Big5`	Παραδοσιακά Κινέζικα (κωδικοποίηση Big5)

Κάποια έγγραφα μπορεί να μην είναι διαθέσιμα σε όλες τις γλώσσες.

- Τη μορφή του εγγράφου. Παράγουμε τεκμηρίωση σε πλήθος διαφορετικών μορφών εξόδου. Κάθε μορφή έχει τα δικά της πλεονεκτήματα και μειονεκτήματα. Κάποιες μορφές είναι περισσότερο κατάλληλες για ανάγνωση online, ενώ άλλες δίνουν πιο καλαίσθητο αποτέλεσμα όταν εκτυπωθούν. Η διάθεση της τεκμηρίωσης σε όλες αυτές τις μορφές εξασφαλίζει ότι οι αναγνώστες μας θα μπορούν να διαβάσουν τα τμήματα που τους ενδιαφέρουν, είτε στην οθόνη τους, είτε αφού τα εκτυπώσουν. Οι διαθέσιμες αυτή τη στιγμή μορφές είναι:

Μορφή	Ερμηνεία
`html-split`	Συλλογή μικρών, συνδεμένων μεταξύ τους, αρχείων HTML.
`html`	Ένα μεγάλο HTML αρχείο που περιέχει ολόκληρο το έγγραφο.
`pdb`	Μορφή βάσης δεδομένων για το Palm Pilot, για χρήση με το πρόγραμμα ανάγνωσης iSilo.
`pdf`	Μορφή κειμένου PDF της Adobe.
`ps`	PostScript®
`rtf`	Μορφή Εμπλουτισμένου Κειμένου της Microsoft
`txt`	Απλό κείμενο

- Ο τρόπος συμπίεσης και πακεταρίσματος. Υπάρχουν τρεις τρόποι που χρησιμοποιούνται αυτή τη στιγμή.

  a. Όταν η μορφή είναι `html-split`, τα αρχεία συμπιέζονται με χρήση της tar(1). Το αρχείο .tar που προκύπτει, συμπιέζεται έπειτα με τους τρόπους συμπίεσης που περιγράφονται παρακάτω.

  b. Όλες οι άλλες μορφές δημιουργούν ένα αρχείο που ονομάζεται book.μορφή (π.χ., book.pdb, book.html, κ.ο.κ.).

     Τα αρχεία αυτά συμπιέζονται κατόπιν με δύο τρόπους συμπίεσης.

Τρόπος	Περιγραφή
`zip`	Μορφή συμπίεσης Zip. Αν θέλετε να το αποσυμπιέσετε στο FreeBSD θα πρέπει να εγκαταστήσετε πρώτα το port archivers/unzip.
`bz2`	Η μορφή BZip2. Είναι λιγότερο διαδεδομένη από το Zip, αλλά γενικά δημιουργεί μικρότερα αρχεία. Εγκαταστήστε το port archivers/bzip2 για να αποσυμπιέσετε αρχεία αυτού του τύπου.

Με τον τρόπο αυτό, η μορφή PostScript® του Εγχειριδίου, συμπιεσμένη με χρήση του BZip2 θα αποθηκευτεί σε ένα αρχείο με όνομα book.ps.bz2 στον κατάλογο handbook/.

Αφού επιλέξετε τη μορφή και το μηχανισμό συμπίεσης που επιθυμείτε να κατεβάσετε, θα πρέπει έπειτα να αποφασίσετε αν θέλετε ή όχι να κατεβάσετε το έγγραφο με τη μορφή *πακέτου* του FreeBSD.

Το πλεονέκτημα στο να κατεβάσετε και να εγκαταστήσετε το πακέτο είναι ότι έπειτα μπορείτε να διαχειριστείτε την τεκμηρίωση χρησιμοποιώντας τα συνηθισμένα εργαλεία διαχείρισης πακέτων του FreeBSD όπως την pkg_add(1) και την pkg_delete(1).

Αν αποφασίσετε να κατεβάσετε και να εγκαταστήσετε την τεκμηρίωση ως πακέτο, θα πρέπει να ξέρετε το ακριβές όνομα αρχείου που θα κατεβάσετε. Τα αρχεία τεκμηρίωσης-ως-πακέτο αποθηκεύονται σε ένα κατάλογο με το όνομα packages. Κάθε πακέτο μοιάζει με όνομα-κειμένου.γλώσσα.κωδικοποίηση.μορφή.tgz.

Για παράδειγμα, το FAQ, στα Αγγλικά, σε μορφή PDF, είναι στο πακέτο με όνομα faq.en_US.ISO8859-1.pdf.tgz.

Αν το ξέρετε αυτό μπορείτε να χρησιμοποιήσετε την ακόλουθη εντολή για να εγκαταστήσετε το πακέτο του Αγγλικού PDF FAQ:

```
# pkg_add ftp://ftp.FreeBSD.org/pub/FreeBSD/doc/packages/faq.en_US.ISO8859-1.pdf.tgz
```

Αφού το κάνετε αυτό, μπορείτε να χρησιμοποιήσετε την εντολή pkg_info(1) για να βρείτε που έχει εγκατασταθεί το αρχείο.

```
# pkg_info -f faq.en_US.ISO8859-1.pdf
Information for faq.en_US.ISO8859-1.pdf:

Packing list:
    Package name: faq.en_US.ISO8859-1.pdf
    CWD to /usr/shared/doc/en_US.ISO8859-1/books/faq
File: book.pdf
    CWD to .
File: +COMMENT (ignored)
```

```
File: +DESC (ignored)
```

Όπως μπορείτε να δείτε, το book.pdf θα έχει εγκατασταθεί στον κατάλογο /usr/shared/doc/en_US.ISO8859-1/books/faq.

Αν δεν θέλετε να χρησιμοποιήσετε τα πακέτα, θα πρέπει να κατεβάσετε μόνοι σας τα συμπιεσμένα αρχεία, να τα αποσυμπιέσετε και έπειτα να αντιγράψετε τα αντίστοιχα έγγραφα στη θέση τους.

Για παράδειγμα, η έκδοση του FAQ σε συνδεδεμένα αρχεία HTML, συμπιεσμένη με χρήση του bzip2(1), μπορεί να βρεθεί στο αρχείο doc/en_US.ISO8859-1/books/faq/book.html-split.tar.bz2. Για να κατεβάσετε και να αποσυμπιέσετε αυτό το αρχείο θα πρέπει να κάνετε το παρακάτω:

```
# fetch ftp://ftp.FreeBSD.org/pub/FreeBSD/doc/en_US.ISO8859-1/books/faq/book.html-
split.tar.bz2
# bzip2 -d book.html-split.tar.bz2
# tar xvf book.html-split.tar
```

Θα καταλήξετε με μια συλλογή από αρχεία .html. Το βασικό ονομάζεται index.html, και θα περιέχει τον πίνακα περιεχομένων, εισαγωγικό υλικό, και δεσμούς προς τα άλλα τμήματα του εγγράφου. Μπορείτε έπειτα να αντιγράψετε ή να μετακινήσετε τα αρχεία αυτά στις τελικές τους θέσεις, όπως απαιτείται.

2.3. Που μπορώ να βρω πληροφορίες για λίστες ηλεκτρονικού ταχυδρομείου του FreeBSD;

Μπορείτε να βρείτε πλήρεις πληροφορίες στην αντίστοιχη καταχώρηση του Εγχειριδίου για τις λίστες ηλεκτρονικού ταχυδρομείου.

2.4. Τι ομάδες συζήτησης είναι διαθέσιμες για το FreeBSD;

Μπορείτε να βρείτε πλήρεις πληροφορίες στην καταχώρηση του Εγχειριδίου για τις ομάδες συζητήσεων.

2.5. Υπάρχουν κανάλια στο IRC (Internet Relay Chat) για το FreeBSD;

Ναι, τα περισσότερα IRC δίκτυα διαθέτουν κανάλι συζήτησης για το FreeBSD:

- Το κανάλι `#FreeBSD` στο EFNet ασχολείται με το FreeBSD, αλλά μην πάτε εκεί για τεχνική υποστήριξη ή για να προσπαθήσετε να πείσετε τους θαμώνες να σας βοηθήσουν για να απαλλαγείτε από τον κόπο να διαβάσετε τις σελίδες βοήθειας ή να ερευνήσετε μόνος σας. Είναι απλώς ένα κανάλι συζήτησης, πρώτα από όλα, και τα θέματα εκεί μπορεί να κυμαίνονται ανάμεσα στο σεξ, τον αθλητισμό, τα πυρηνικά όπλα ως και το FreeBSD. Σας έχουμε προειδοποιήσει! Είναι διαθέσιμο στον εξυπηρετητή `irc.chat.org`.

- Το κανάλι #FreeBSDhelp στο EFNet είναι αφιερωμένο στη βοήθεια των χρηστών του FreeBSD. Έχετε πολύ μεγαλύτερη πιθανότητα να βρείτε απαντήσεις στις ερωτήσεις σας σε αυτό, παρά στο #FreeBSD.
- Το κανάλι FreeBSD στο Freenode είναι ένα γενικό κανάλι βοήθειας με περίπου 300 χρήστες κάθε φορά. Οι συζητήσεις μερικές φορές ξεφεύγουν από το θέμα, αλλά δίνεται προτεραιότητα στους χρήστες με ερωτήσεις για το FreeBSD. Είμαστε καλοί στο να σας βοηθήσουμε να καταλάβετε τα βασικά, να σας δείξουμε που να ανατρέξετε στο Εγχειρίδιο όποτε χρειάζεται, και να σας κατευθύνουμε εκεί όπου μπορείτε να μάθετε περισσότερα για θέμα που σας απασχολεί. Είμαστε κατά βάση Αγγλόφωνο κανάλι, αν και έχουμε χρήστες από όλο τον κόσμο. Αν θέλετε να ρωτήσετε στη μητρική σας γλώσσα, προσπαθήστε να κάνετε την ερώτηση στα Αγγλικά και έπειτα μετακινηθείτε σε κάποιο άλλο κανάλι freebsd-*lang* αν χρειάζεται.
- Το κανάλι #FreeBSD στο DALNET είναι διαθέσιμο στο irc.dal.net στις ΗΠΑ και στο irc.eu.dal.net στην Ευρώπη.
- Το κανάλι #FreeBSDHelp στο DALNET είναι διαθέσιμο στο irc.dal.net στις ΗΠΑ και στο irc.eu.dal.net στην Ευρώπη.
- Το κανάλι #FreeBSD στο UNDERNET είναι διαθέσιμο στο us.undernet.org στις ΗΠΑ και στο eu.undernet.org στην Ευρώπη. Καθώς είναι κανάλι βοήθειας, θα πρέπει να είστε προετοιμασμένοι να διαβάσετε τα έγγραφα στα οποία θα σας παραπέμψουν.
- Το κανάλι #FreeBSD στο RUSNET είναι προσανατολισμένο στη βοήθεια χρηστών του FreeBSD που μιλούν τη Ρωσική γλώσσα. Είναι επίσης ένα καλό μέρος για μη-τεχνικές συζητήσεις.
- Το κανάλι #BSDChat στο IRCNET είναι ένα κανάλι σε παραδοσιακή Κινεζική γλώσσα, αφιερωμένο στη βοήθεια χρηστών του FreeBSD. Είναι επίσης ένα καλό μέρος για μη-τεχνικές συζητήσεις.

Κάθε ένα από αυτά τα κανάλια, είναι χωριστό, και δεν διασυνδέεται με τα υπόλοιπα. Τα στυλ της συζήτησης διαφέρουν, και ίσως θα θέλετε να δοκιμάσετε καθένα για να βρείτε κάποιο που να ταιριάζει στο δικό σας. Όπως με *κάθε* τύπο συζήτησης στο IRC, αν είστε από αυτούς που προσβάλλονται εύκολα, ή δεν μπορείτε να αντιμετωπίσετε πολλούς νέους ανθρώπους (και αρκετούς πιο μεγάλους) να επιδίδονται στο λεκτικό αντίστοιχο της πυγμαχίας, μην ασχοληθείτε καθόλου.

2.6. Που μπορώ να βρω επί πληρωμή υποστήριξη και εκπαίδευση για το FreeBSD;

Το DaemonNews παρέχει επί πληρωμή εκπαίδευση και υποστήριξη για το FreeBSD. Περισσότερες πληροφορίες μπορείτε να βρείτε στην δικτυακή τους τοποθεσία BSD Mall.

Το FreeBSD Mall παρέχει επίσης επί πληρωμή υποστήριξη για το FreeBSD. Μπορείτε να βρείτε περισσότερες πληροφορίες στην δικτυακή τους τοποθεσία.

Οποιοσδήποτε άλλος οργανισμός παρέχει εκπαίδευση και υποστήριξη, θα πρέπει να επικοινωνήσει με το Project για να καταχωρηθεί σε αυτή τη λίστα.

Chapter 3. Εγκατάσταση

3.1. Ποιο αρχείο πρέπει να κατεβάσω για να πάρω το FreeBSD;

Χρειάζεστε τρία images δισκετών: floppies/boot.flp, floppies/kern1.flp, και floppies/kern2.flp. Τα images αυτά πρέπει να να τα γράψετε σε δισκέτες χρησιμοποιώντας κάποιο εργαλείο όπως το `fdimage` ή το dd(1).

Αν πρέπει να κατεβάσετε μόνοι σας τα σετ διανομών (distributions - για παράδειγμα για μια εγκατάσταση μέσω συστήματος αρχείων DOS), θα βρείτε παρακάτω μερικές συστάσεις σχετικά με το ποιες διανομές να κατεβάσετε:

- base/
- manpages/
- compat*/
- doc/
- src/ssys.*

Πλήρεις οδηγίες σχετικά με τη διαδικασία καθώς και περισσότερες πληροφορίες σχετικά με γενικά προβλήματα κατά την εγκατάσταση μπορείτε να βρείτε στο Κεφάλαιο του Εγχειριδίου σχετικά με την εγκατάσταση του FreeBSD.

3.2. Τι μπορώ να κάνω αν το image της δισκέτας δεν χωράει σε μια μόνο δισκέτα;

Μια δισκέτα των 3.5 ιντσών (1.44MB) μπορεί να χωρέσει ως 1474560 bytes δεδομένων. Το αρχείο image της δισκέτας εκκίνησης είναι ακριβώς αυτό το μέγεθος.

Συνηθισμένα λάθη κατά την προετοιμασία της δισκέτας εκκίνησης:

- Κατέβασμα της δισκέτας χωρίς τη χρήση του *δυαδικού (binary)* τρόπου όταν χρησιμοποιείται εξυπηρετητής FTP.

  Κάποια προγράμματα-πελάτες για FTP, χρησιμοποιούν από προεπιλογή τον *ascii (κειμένου)* τρόπο μεταφοράς δεδομένων και προσπαθούν να αλλάξουν τους χαρακτήρες τέλους γραμμής ώστε να ταιριάζουν με τις συμβάσεις που χρησιμοποιούνται από το σύστημα στο οποίο γίνεται η λήψη. Αυτό σχεδόν σίγουρα θα καταστρέψει το image εκκίνησης. Ελέγξτε το μέγεθος του image εκκίνησης που κατεβάσατε: Αν δεν είναι *ακριβώς* αυτό που φαίνεται στον εξυπηρετητή, η διαδικασία του κατεβάσματος θα πρέπει να θεωρηθεί ύποπτη.

  Για να παρακάμψετε το πρόβλημα: γράψτε *binary* στην γραμμή εντολών του FTP, μετά που θα συνδεθείτε στον εξυπηρετητή, και πριν ξεκινήσετε να κατεβάζετε το image.

- Χρήση της εντολής `copy` του DOS (η αντίστοιχου γραφικού εργαλείου) για τη μεταφορά του

αρχείου image στη δισκέτα.

Προγράμματα όπως το `copy`, δεν θα λειτουργήσουν σωστά καθώς το image εκκίνησης έχει φτιαχτεί ώστε να εκκινεί άμεσα. Το image έχει τα πλήρη περιεχόμενα της δισκέτας, τροχιά προς τροχιά, και δεν πρέπει να γραφεί στη δισκέτα ως κανονικό αρχείο. Θα πρέπει να το μεταφέρετε στη δισκέτα με "ωμό" τρόπο, χρησιμοποιώντας εργαλεία χαμηλού επιπέδου (low-level) (όπως π.χ. το `fdimage` ή το `rawrite`) που περιγράφονται στον οδηγό εγκατάστασης του FreeBSD.

3.3. Που βρίσκονται οι οδηγίες για την εγκατάσταση του FreeBSD;

Μπορείτε να βρείτε τις οδηγίες στην καταχώρηση του Εγχειριδίου σχετικά με την εγκατάσταση του FreeBSD.

3.4. Τι χρειάζομαι για να εκτελέσω το FreeBSD;

Για εκδόσεις του FreeBSD από το 5.X και μετά χρειάζεστε ένα PC με επεξεργαστή 486 ή καλύτερο, με 24 MB ή περισσότερη RAM, και τουλάχιστον 150 MB χώρου στο σκληρό σας δίσκο.

Μπορείτε να εκτελέσετε όλες τις εκδόσεις του FreeBSD με μια χαμηλών προδιαγραφών, μονόχρωμη (MDA) κάρτα γραφικών, αλλά για να χρησιμοποιήσετε το Xorg, θα χρειαστείτε κάρτα VGA ή καλύτερη.

Δείτε επίσης και το Συμβατότητα Υλικού.

3.5. Πως μπορώ να δημιουργήσω τη δική μου, προσαρμοσμένη δισκέτα εγκατάστασης;

Τη δεδομένη στιγμή, δεν υπάρχει τρόπος να φτιάξετε *απλώς* μια προσαρμοσμένη δισκέτα εγκατάστασης. Θα πρέπει να δημιουργήσετε ολόκληρη νέα έκδοση, η οποία θα περιλαμβάνει και την προσαρμοσμένη δισκέτα εγκατάστασης σας.

Για να δημιουργήσετε μια προσαρμοσμένη έκδοση, ακολουθήστε τις οδηγίες στο άρθρο σχετικά με την Οργάνωση Εκδόσεων.

3.6. Μπορούν να συνυπάρξουν τα Windows® με το FreeBSD;

Εγκαταστήστε πρώτα τα Windows®, και μετά το FreeBSD. Ο διαχειριστής εκκίνησης του FreeBSD θα σας επιτρέπει έπειτα να επιλέξετε την εκκίνηση είτε των Windows® είτε του FreeBSD. Αν εγκαταστήσετε τα Windows® μετά το FreeBSD, θα σας σβήσουν τον διαχειριστή εκκίνησης, χωρίς καν να σας ρωτήσουν. Αν σας συμβεί, αυτό δείτε το επόμενο τμήμα.

3.7. Τα Windows® κατέστρεψαν το διαχειριστή εκκίνησης μου! Πως θα τον επαναφέρω;

Μπορείτε να επανεγκαταστήσετε το διαχειριστή εκκίνησης του FreeBSD χρησιμοποιώντας έναν από τους τρεις παρακάτω τρόπους:

- Εκτελώντας το DOS, μετακινηθείτε στον κατάλογο tools/ της FreeBSD διανομής σας, και ψάξτε για το αρχείο bootinst.exe. Εκτελέστε το όπως φαίνεται παρακάτω:

  ```
  ...\TOOLS> bootinst.exe boot.bin
  ```

  και ο διαχειριστής εκκίνησης θα επανεγκατασταθεί.

- Ξεκινήστε ξανά χρησιμοποιώντας τη δισκέτα εκκίνησης του FreeBSD και πηγαίνετε στην επιλογή Custom Installation του μενού. Επιλέξτε Partition. Επιλέξτε τον οδηγό που κανονικά θα περιείχε τον διαχειριστή εκκίνησης σας (φυσιολογικά τον πρώτο) και όταν φτάσετε στον επεξεργαστή κατατμήσεων (partition editor) για τον οδηγό αυτό, το πρώτο πράγμα που θα κάνετε (χωρίς άλλες αλλαγές), είναι να επιλέξετε (W)rite. Επιλέξτε yes στην επιβεβαίωση που θα εμφανιστεί και όταν φτάσετε στην προτροπή επιλογής Διαχειριστή Εκκίνησης, επιλέξτε "Boot Manager". Με τον τρόπο αυτό ο διαχειριστής εκκίνησης θα γραφεί ξανά στο δίσκο. Μπορείτε τώρα να βγείτε από το μενού της εγκατάστασης και να επανεκκινήσετε από το σκληρό δίσκο, όπως συνήθως.

- Ξεκινήστε με τη βοήθεια της δισκέτας εκκίνησης (ή του CD) του FreeBSD και επιλέξτε "Fixit" από το μενού. Επιλέξτε είτε τη δισκέτα Fixit είτε το CD #2 (το "live" σύστημα αρχείων) και θα εισέλθετε στο κέλυφος fixit. Εκτελέστε έπειτα την ακόλουθη εντολή:

  ```
  Fixit# fdisk -B -b /boot/boot0 bootdevice
  ```

  αντικαθιστώντας το *bootdevice* με τον πραγματικό σας δίσκο εκκίνησης, όπως π.χ. ad0 (πρώτος δίσκος IDE), ad4 (πρώτος δίσκος IDE στον βοηθητικό ελεγκτή), da0 (πρώτος δίσκος SCSI), κ.λ.π.

3.8. Ο φορητός μου υπολογιστής σειράς IBM Thinkpad, σειράς A, T ή X, σταματάει να αποκρίνεται όταν προσπαθώ να εκκινήσω το FreeBSD μετά την εγκατάσταση. Πως μπορώ να επιλύσω αυτό το πρόβλημα;

Πρόκειται για ένα πρόβλημα στις αρχικές εκδόσεις του BIOS της IBM στα συγκεκριμένα μηχανήματα, το οποίο αναγνωρίζει την κατάτμηση του FreeBSD ως πιθανό διαμέρισμα FAT για τη λειτουργία αδρανοποίησης στο δίσκο (suspend-to-disk). Το BIOS σταματάει να αποκρίνεται καθώς προσπαθεί να αναλύσει το διαμέρισμα του FreeBSD.

Σύμφωνα με την IBM, τα ακόλουθα μοντέλα και εκδόσεις BIOS περιλαμβάνουν τις απαραίτητες

διορθώσεις.

Μοντέλο	Έκδοση BIOS
T20	IYET49WW ή μεταγενέστερο
T21	KZET22WW ή μεταγενέστερο
A20p	IVET62WW ή μεταγενέστερο
A20m	IWET54WW ή μεταγενέστερο
A21p	KYET27WW ή μεταγενέστερο
A21m	KXET24WW ή μεταγενέστερο
A21e	KUET30WW

Έχει αναφερθεί ότι μεταγενέστερες εκδόσεις του BIOS της IBM, ίσως παρουσιάζουν ξανά το πρόβλημα. Το μήνυμα αυτό από τον Jacques Vidrine στην ηλεκτρονική λίστα του FreeBSD για φορητούς υπολογιστές περιγράφει μια διαδικασία η οποία ίσως δουλέψει σε νεότερα μοντέλα φορητών της IBM τα οποία δεν εκκινούν κανονικά το FreeBSD, και στα οποία μπορείτε να αναβαθμίσετε το BIOS ή και να επιστρέψετε σε προηγούμενη έκδοση του.

Αν έχετε παλιότερο BIOS, αλλά δεν έχετε την επιλογή να το αναβαθμίσετε, ένας τρόπος να ξεπεράσετε το πρόβλημα είναι να εγκαταστήσετε το FreeBSD, να αλλάξετε τον αναγνωριστικό αριθμό (partition id) της κατάτμησης, και να εγκαταστήσετε νέα boot blocks τα οποία να μπορούν να χειριστούν το διαφορετικό αναγνωριστικό της κατάτμησης.

Αρχικά, θα πρέπει να επαναφέρετε το μηχάνημα σε μια κατάσταση ώστε να περνάει από την αρχική διαγνωστική οθόνη. Αυτό απαιτεί να ενεργοποιήσετε το μηχάνημα χωρίς να το αφήσετε να βρει την κατάτμηση του FreeBSD στον κύριο δίσκο του. Ένας τρόπος είναι να απομακρύνετε τον σκληρό δίσκο και να τον μετακινήσετε προσωρινά σε ένα παλιότερο ThinkPad (όπως το ThinkPad 600) ή κάποιο επιτραπέζιο μηχάνημα, χρησιμοποιώντας το κατάλληλο καλώδιο μετατροπής. Στο μηχάνημα αυτό μπορείτε πλέον να διαγράψετε την κατάτμηση του FreeBSD και έπειτα να μετακινήσετε τον δίσκο στο αρχικό μηχάνημα. Το ThinkPad θα πρέπει τώρα να εκκινεί κανονικά.

Με το μηχάνημα σε λειτουργική κατάσταση, μπορείτε τώρα να χρησιμοποιήσετε την διαδικασία που περιγράφεται εδώ ώστε να έχετε τελικά μια εγκατάσταση του FreeBSD που να λειτουργεί.

1. Κατεβάστε τις δισκέτες boot1 και boot2 από http://people.FreeBSD.org/~bmah/ThinkPad/. Αποθηκεύστε τα αρχεία αυτά κάπου που θα μπορείτε να τα ξαναβρείτε αργότερα.
2. Εγκαταστήστε κανονικά το FreeBSD στο ThinkPad. *Μην* χρησιμοποιήσετε την κατάσταση `Dangerously Dedicated`. *Μην* κάνετε επανεκκίνηση όταν τελειώσει η εγκατάσταση.
3. Χρησιμοποιήστε είτε το "Emergency Holographic Shell" (`ALT` + `F4`) είτε το κέλυφος "fixit".
4. Χρησιμοποιήστε την fdisk(8) για να αλλάξετε τον αναγνωριστικό αριθμό της κατάτμησης του FreeBSD από `165` σε `166` (αυτός είναι ο τύπος που χρησιμοποιείται από το OpenBSD).
5. Γράψτε τα αρχεία boot1 και boot2 στο τοπικό σύστημα αρχείων.
6. Χρησιμοποιήστε το disklabel(8) για να γράψετε τα boot1 και boot2 στο slice του FreeBSD.

```
# disklabel -B -b boot1 -s boot2 ad0sn
```

Το *n* είναι ο αριθμός του slice που έχετε εγκαταστήσει το FreeBSD.

7. Κάντε επανεκκίνηση. Στην προτροπή εκκίνησης θα δείτε την επιλογή να ξεκινήσετε το OpenBSD. Στην πραγματικότητα, με αυτό τον τρόπο θα ξεκινήσετε το FreeBSD.

Το να κάνετε το παραπάνω να λειτουργήσει στην περίπτωση που θέλετε να έχετε διπλή εκκίνηση OpenBSD και FreeBSD στον ίδιο φορητό υπολογιστή, το αφήνουμε ως άσκηση στον αναγνώστη.

3.9. Μπορώ να κάνω εγκατάσταση σε ένα δίσκο με χαλασμένους τομείς;

Μπορείτε, αλλά είναι κακή ιδέα.

Αν δείτε χαλασμένους τομείς σε ένα σύγχρονο οδηγό IDE, υπάρχει μεγάλη πιθανότητα ο οδηγός αυτός, πολύ σύντομα, να σταματήσει να λειτουργεί εντελώς (ο οδηγός δεν έχει άλλους εναλλακτικούς κυλίνδρους που να μπορούν να αντικαταστήσουν τους χαλασμένους μέσω της εσωτερικής λειτουργίας επανατοποθέτησης των δεδομένων, κάτι το οποίο σημαίνει ότι ο δίσκος έχει σημαντική φθορά). Σας συνιστούμε να αγοράσετε καινούριο δίσκο.

Αν έχετε οδηγό SCSI με χαλασμένους τομείς δείτε αυτή την απάντηση.

3.10. Διάφορα περίεργα πράγματα συμβαίνουν όταν εκκινώ με τη δισκέτα εκκίνησης! Τι συμβαίνει;

Αν βλέπετε καταστάσεις όπως το μηχάνημα να μην αποκρίνεται ή να κάνει επανεκκίνηση μόνο του όταν προσπαθείτε να εκκινήσετε με την δισκέτα εκκίνησης, θα πρέπει να κάνετε στον εαυτό σας τρεις ερωτήσεις:

1. Χρησιμοποιήσατε καινούριες, φρεσκο-διαμορφωμένες, και χωρίς λάθη δισκέτες (κατά προτίμηση καλής εταιρίας τις οποίες μόλις βγάλατε καινούριες από το κουτί τους, σε αντίθεση με τη δισκέτα που πήρατε μαζί με κάποιο περιοδικό και η οποία βρίσκονταν κάτω από το κρεβάτι τα τρία τελευταία χρόνια);
2. Κατεβάσατε το image της δισκέτας με δυαδικό (binary ή image) τρόπο μεταφοράς; (μη ντρέπεστε, ακόμα και οι καλύτεροι από μας έχουν κατά λάθος κατεβάσει κάποιο δυαδικό αρχείο σε κατάσταση ASCII (κειμένου), τουλάχιστον μια φορά!)
3. Αν χρησιμοποιείτε Windows® 95 ή 98, εκτελέστε το `fdimage` ή το `rawrite` σε καθαρή κατάσταση DOS; Τα λειτουργικά αυτά μπορεί να παρεμβληθούν σε προγράμματα τα οποία γράφουν απευθείας στο υλικό, κάτι το οποίο συμβαίνει και με τα προγράμματα δημιουργίας των δισκετών. Μπορεί να δημιουργηθεί πρόβλημα, ακόμα και αν τα εκτελείτε σε παράθυρο DOS μέσα από το γραφικό περιβάλλον.

Έχουν επίσης αναφερθεί περιπτώσεις όπου το getenv(3) δημιουργεί προβλήματα στο κατέβασμα της δισκέτας εκκίνησης, έτσι είναι καλύτερα να χρησιμοποιήσετε κάποιο άλλο πρόγραμμα FTP, αν αυτό

είναι δυνατόν.

3.11. Ξεκίνησα από το ATAPI CDROM μου, αλλά το πρόγραμμα εγκατάστασης λέει ότι δεν βρίσκει CDROM. Που πήγε;

Η συνήθης αιτία αυτού του προβλήματος είναι ένας κακορυθμισμένος οδηγός CDROM. Πολλά PC έρχονται πλέον με το CDROM ως slave συσκευή στο δευτερεύοντα ελεγκτή, χωρίς να υπάρχει στο ίδιο κανάλι συσκευή master. Αυτό, σύμφωνα με τις προδιαγραφές του ATAPI, δεν είναι έγκυρο, αλλά τα Windows® τηρούν τις προδιαγραφές χαλαρά, ενώ το BIOS το αγνοεί κατά την εκκίνηση. Αυτός είναι και ο λόγος που το BIOS κατάφερε να δει το CDROM και να το χρησιμοποιήσει, αλλά και ο λόγος που το FreeBSD δεν μπόρεσε να το δει για να συνεχίσει την εγκατάσταση.

Ρυθμίστε ξανά το σύστημα σας, ώστε το CDROM είτε να είναι η master συσκευή στον ελεγκτή που είναι συνδεμένη, ή βεβαιωθείτε ότι είναι slave σε ένα ελεγκτή IDE ο οποίος όμως έχει άλλη συσκευή συνδεμένη ως master.

3.12. Μπορώ να εγκαταστήσω το FreeBSD στο φορητό μου χρησιμοποιώντας PLIP (Parallel Line IP, IP μέσω Παράλληλης Θύρας);

Ναι. Χρησιμοποιήστε τυποποιημένο καλώδιο τύπου Laplink. Αν χρειάζεται, διαβάστε το τμήμα PLIP του Εγχειριδίου για λεπτομέρειες σχετικές με δικτύωση μέσω παράλληλης θύρας.

3.13. Τι γεωμετρία να χρησιμοποιήσω για το σκληρό μου δίσκο;

> Με τον όρο "γεωμετρία", εννοούμε τον αριθμό των κυλίνδρων, κεφαλών και τομέων ανά τροχιά ενός δίσκου. Για ευκολία, θα αναφερόμαστε στον όρο αυτό ως C/H/S (Cylinders / Heads / Sectors). Αυτός είναι και ο τρόπος με τον οποίο το BIOS βρίσκει σε ποια περιοχή του δίσκου να γράψει.

Το παραπάνω προκαλεί σύγχυση στου νέους διαχειριστές συστημάτων. Καταρχήν, η *φυσική* γεωμετρία ενός οδηγού SCSI είναι συνολικά άσχετη, αφού το FreeBSD λειτουργεί με βάση τα μπλοκ δίσκου. Στην πραγματικότητα, δεν υπάρχει καν "αληθινή" φυσική γεωμετρία, καθώς η πυκνότητα των τομέων μεταβάλλεται από περιοχή σε περιοχή του δίσκου. Αυτό που οι κατασκευαστές αναφέρουν συχνά ως "φυσική γεωμετρία" είναι συνήθως η γεωμετρία με την όποια έχουν ανακαλύψει ότι γίνεται η μικρότερη σπατάλη ελεύθερου χώρου. Για δίσκους IDE, το FreeBSD χρησιμοποιεί πράγματι το C/H/S, αλλά όλοι οι μοντέρνοι οδηγοί μετατρέπουν εσωτερικά αυτές τις αναφορές σε αντίστοιχα μπλοκ.

Το μόνο που έχει πραγματική σημασία είναι η *λογική* γεωμετρία. Αυτή είναι και η απάντηση που δέχεται το BIOS όταν ρωτάει το δίσκο "ποια είναι η γεωμετρία σου;" Έπειτα, χρησιμοποιεί αυτή τη γεωμετρία για να αποκτήσει πρόσβαση στο δίσκο. Καθώς το FreeBSD χρησιμοποιεί το BIOS κατά

την εκκίνηση, είναι πολύ σημαντικό η γεωμετρία αυτή να είναι σωστή. Ειδικότερα, αν έχετε περισσότερα από ένα λειτουργικά συστήματα στο δίσκο, πρέπει όλα να συμφωνούν όσο αφορά τη γεωμετρία. Διαφορετικά θα έχετε σοβαρά προβλήματα κατά την εκκίνηση!

Για δίσκους SCSI, η γεωμετρία που πρέπει να χρησιμοποιηθεί, εξαρτάται από το αν είναι ενεργοποιημένη η υποστήριξη εκτεταμένης μετάφρασης (αυτό συχνά αναφέρεται ως "υποστήριξη για δίσκους DOS 1GB" ή κάτι αντίστοιχο). Αν είναι απενεργοποιημένη, χρησιμοποιήστε *N* κυλίνδρους, 64 κεφαλές και 32 τομείς/τροχιά, όπου το *N* είναι η χωρητικότητα του δίσκου σε MB. Για παράδειγμα, για ένα δίσκο 2GB, θα πρέπει να έχετε 2048 κυλίνδρους, 64 κεφαλές, και 32 τομείς/τροχιά.

Αν *είναι* ενεργοποιημένη (και συχνά παρέχεται έτσι ώστε να ξεπερνιούνται κάποιοι περιορισμοί στο MS-DOS®) και η χωρητικότητα του δίσκου είναι μεγαλύτερη από 1GB, χρησιμοποιήστε M κυλίνδρους, 63 τομείς ανά τροχιά (*όχι* 64) και 255 κεφαλές, όπου `M` είναι η χωρητικότητα του δίσκου σε MB, διαιρεμένη με το 7.844238 (!). Έτσι, στο παράδειγμα μας, ο δίσκος των 2GB θα είχε 261 κυλίνδρους, 63 τομείς ανά τροχιά και 255 κεφαλές.

Αν δεν είστε σίγουρος για το παραπάνω, ή αν το FreeBSD αποτύχει στην ανίχνευση της σωστής γεωμετρίας κατά την εγκατάσταση, ο απλούστερος τρόπος για να το παρακάμψετε, είναι συνήθως να δημιουργήσετε ένα μικρό διαμέρισμα DOS στο δίσκο. Το BIOS έπειτα θα ανιχνεύσει τη σωστή γεωμετρία, και μπορείτε πάντα να διαγράψετε το διαμέρισμα DOS μέσα από τον επεξεργαστή κατατμήσεων, αν δε θέλετε να το κρατήσετε. Μπορεί ωστόσο να θελήσετε να το αφήσετε, για να προγραμματίζετε κάρτες δικτύου και για άλλες, αντίστοιχες, εργασίες.

Εναλλακτικά, υπάρχει ένα ελεύθερα διαθέσιμο βοηθητικό πρόγραμμα το οποίο διανέμεται με το FreeBSD και λέγεται pfdisk.exe. Μπορείτε να το βρείτε στον υποκατάλογο tools στο CDROM του FreeBSD ή στις δικτυακές τοποθεσίες FTP του FreeBSD. Το πρόγραμμα αυτό μπορεί να χρησιμοποιηθεί για να ανακαλύψει τι γεωμετρία χρησιμοποιείται από τα άλλα λειτουργικά συστήματα στο δίσκο που χρησιμοποιείται. Μπορείτε να εισάγετε απευθείας αυτή τη γεωμετρία στον επεξεργαστή κατατμήσεων.

3.14. Υπάρχουν κάποιοι περιορισμοί στο πως πρέπει να χωρίσω το δίσκο;

Ναι. Πρέπει να βεβαιωθείτε ότι η ριζική (root) κατάτμηση βρίσκεται κάτω από τους 1024 κυλίνδρους, ώστε το BIOS να μπορεί να εκκινήσει τον πυρήνα από αυτή. (Σημειώστε ότι αυτός είναι ένας περιορισμός στο BIOS του PC, και όχι στο FreeBSD).

Για ένα δίσκο SCSI, συνήθως αυτό σημαίνει ότι η ριζική κατάτμηση θα βρίσκεται στα πρώτα 1024MB (ή στα πρώτα 4096MB αν χρησιμοποιείται η εκτεταμένη μετάφραση - δείτε την προηγούμενη ερώτηση). Για δίσκους IDE, το αντίστοιχο όριο είναι τα 504MB.

3.15. Είναι συμβατό το FreeBSD με προγράμματα διαχείρισης δίσκων (disk managers);

Το FreeBSD αναγνωρίζει και επιτρέπει τη χρήση του Ontrack Disk Manager. Δεν υποστηρίζονται άλλοι διαχειριστές δίσκων.

Αν θέλετε απλώς να χρησιμοποιήσετε το δίσκο με το FreeBSD, δεν χρειάζεστε διαχειριστή δίσκου. Απλώς ρυθμίστε το δίσκο για όσο περισσότερο χώρο μπορεί να δει το BIOS (συνήθως 504MB), και το FreeBSD θα ανακαλύψει πόσο ελεύθερο χώρο έχετε στην πραγματικότητα. Αν χρησιμοποιείτε κάποιο παλιό δίσκο σε ελεγκτή MFM, ίσως να πρέπει να πείτε στο FreeBSD πόσους κυλίνδρους να χρησιμοποιήσει.

Αν θέλετε να χρησιμοποιήσετε το δίσκο τόσο με το FreeBSD όσο και με κάποιο άλλο λειτουργικό σύστημα, θα πρέπει να μπορείτε να το κάνετε χωρίς διαχειριστή δίσκου: απλώς βεβαιωθείτε ότι το διαμέρισμα εκκίνησης του FreeBSD καθώς και η κατάτμηση του άλλου λειτουργικού συστήματος βρίσκονται μέσα στους πρώτους 1024 κυλίνδρους. Αν είστε αρκετά προσεκτικός, ένα διαμέρισμα εκκίνησης (boot) μεγέθους 20MB θα είναι αρκετό.

3.16. Όταν εκκινώ το FreeBSD για πρώτη φορά, παίρνω το μήνυμα Missing Operating System. Τι συμβαίνει;

Αυτή είναι μια κλασική περίπτωση διένεξης μεταξύ του FreeBSD και του DOS ή κάποιου άλλου λειτουργικού σχετικά με την ιδέα που έχει το καθένα για την γεωμετρία του δίσκου. Θα πρέπει να επανεγκαταστήσετε το FreeBSD αλλά τηρώντας προσεκτικά τις οδηγίες που δώσαμε πιο πάνω, είναι σχεδόν σίγουρο ότι θα τα καταφέρετε.

3.17. Γιατί δεν μπορώ να συνεχίσω πέρα από την προτροπή F? του διαχειριστή εκκίνησης;

Αυτό είναι άλλο ένα σύμπτωμα του προβλήματος που περιγράφεται στην προηγούμενη ερώτηση. Δεν συμπίπτει η γεωμετρία του BIOS με αυτή του FreeBSD! Αν ο ελεγκτής ή το BIOS σας υποστηρίζει μετάφραση κυλίνδρων (συχνά αναφέρεται ως "1GB drive support"), δοκιμάστε να αλλάξετε αυτή τη τιμή και να επανεγκαταστήσετε το FreeBSD.

3.18. Πρέπει να εγκαταστήσω όλο τον πηγαίο κώδικα;

Γενικά, όχι. Ωστόσο σας συνιστούμε να εγκαταστήσετε, ως ελάχιστο, τον πηγαίο κώδικα της διανομής `base`, ο οποίος περιλαμβάνει αρκετά από τα αρχεία που αναφέρονται εδώ, καθώς και τον πηγαίο κώδικα της διανομής `sys`, η οποία περιλαμβάνει τον πηγαίο κώδικα του πυρήνα. Δεν υπάρχει ωστόσο κάτι στο σύστημα το οποίο να απαιτεί την ύπαρξη του πηγαίου κώδικα για να λειτουργήσει, εκτός από το πρόγραμμα ρύθμισης πυρήνα config(8). Με εξαίρεση τον πηγαίο κώδικα του πυρήνα, η δομή του συστήματος μεταγλώττισης μας είναι τέτοια, ώστε μπορείτε πάντα να προσαρτήσετε το δέντρο πηγαίου κώδικα μέσω NFS (και με δικαιώματα μόνο ανάγνωσης) και πάλι να μπορείτε να δημιουργήσετε νέα εκτελέσιμα (λόγω του περιορισμού που υπάρχει στον πηγαίο κώδικα του πυρήνα, σας συνιστούμε να μην κάνετε την προσάρτηση απευθείας στον κατάλογο /usr/src, αλλά να χρησιμοποιήσετε κάποια άλλη τοποθεσία και να δημιουργήσετε τους κατάλληλους συμβολικούς δεσμούς που να αντιγράφουν τη δομή της κεντρικής ιεραρχίας του δέντρου πηγαίου κώδικα).

Αν έχετε άμεσα διαθέσιμο τον πηγαίο κώδικα, και γνωρίζετε πως να μεταγλωττίσετε ένα ολόκληρο σύστημα από αυτόν, θα διευκολυνθείτε πάρα πολύ όταν αναβαθμίζετε το σύστημα σας σε μελλοντικές εκδόσεις του FreeBSD.

Για να επιλέξετε ένα υποσύνολο του πηγαίου κώδικα, από την επιλογή Distributions του εργαλείου εγκατάστασης συστήματος, επιλέξτε το μενού Custom.

3.19. Χρειάζεται να φτιάξω προσαρμοσμένο πυρήνα;

Η δημιουργία ενός νέου πυρήνα ήταν αρχικά σχεδόν υποχρεωτικό βήμα σε μια εγκατάσταση FreeBSD, αλλά οι πιο πρόσφατες εκδόσεις έχουν ωφεληθεί από την εισαγωγή αισθητά φιλικότερων προγραμμάτων ρύθμισης του πυρήνα. Από το FreeBSD 5.X και μετά, είναι αρκετά εύκολο να ρυθμίσετε τον πυρήνα χρησιμοποιώντας το πολύ πιο ευέλικτο σύστημα των "hints" τα οποία μπορείτε να ρυθμίσετε στην προτροπή του loader.

Ενδεχομένως να αξίζει ακόμα να δημιουργήσετε ένα νέο πυρήνα ο οποίος να περιέχει μόνο τα προγράμματα οδήγησης που χρειάζεστε, για να γλυτώσετε κάποια μικρή ποσότητα μνήμης RAM, αλλά αυτό δεν είναι πλέον απαραίτητο για τα περισσότερα συστήματα.

3.20. Για τους κωδικούς πρόσβασης των χρηστών, να χρησιμοποιήσω DES, Blowfish, ή MD5 κωδικοποίηση, και πως θα καθορίσω τι θα χρησιμοποιούν οι χρήστες μου;

Η προεπιλεγμένη μορφή κρυπτογράφησης για κωδικούς στο FreeBSD είναι το *MD5*. Η γενική αντίληψη είναι ότι παρέχουν καλύτερη ασφάλεια σε σχέση με την παραδοσιακή μορφή του UNIX® που βασίζεται στον αλγόριθμο *DES*. Οι κωδικοί DES είναι ακόμα διαθέσιμοι, αν χρειάζεται να διαμοιράσετε το αρχείο των κωδικών σας με παλιότερα λειτουργικά συστήματα, τα οποία χρησιμοποιούν ακόμα το παλιότερο και λιγότερο ασφαλές σύστημα (είναι διαθέσιμα αν εγκαταστήσετε τη διανομή "crypto" μέσω του sysinstall ή εγκαθιστώντας τον αντίστοιχο πηγαίο κώδικα αν κάνετε εγκατάσταση μέσω πηγαίου κώδικα). Αν εγκαταστήσετε τις βιβλιοθήκες crypto θα μπορέσετε επίσης να χρησιμοποιήσετε κρυπτογράφηση Blowfish η οποία είναι ακόμα πιο ασφαλής. Το ποια μορφή κωδικών χρησιμοποιείται για τους νέους κωδικούς, ελέγχεται από την δυνατότητα εισόδου "passwd_format" στο /etc/login.conf, το οποίο παίρνει τις τιμές "des", "blf" (αν είναι διαθέσιμη) ή "md5". Δείτε τη σελίδα manual του login.conf(5) για περισσότερες πληροφορίες σχετικά με τις δυνατότητες εισόδου.

3.21. Γιατί ενώ η δισκέτα εκκίνησης ξεκινάει κανονικά, κρεμάει στην οθόνη Probing Devices...;

Αν έχετε εγκατεστημένο οδηγό IDE Zip® ή Jaz®, αφαιρέστε τον και ξαναπροσπαθήστε. Η δισκέτα εκκίνησης μπορεί να μπερδευτεί από αυτούς τους οδηγούς. Μετά την εγκατάσταση του συστήματος, μπορείτε να ξανασυνδέσετε τον οδηγό. Ευελπιστούμε ότι το πρόβλημα αυτό θα διορθωθεί σε επόμενη έκδοση.

3.22. Γιατί παίρνω το μήνυμα λάθους panic: can't mount root όταν εκκινώ το σύστημα για πρώτη φορά μετά την εγκατάσταση;

Το σφάλμα αυτό προέρχεται από την σύγχυση που προκαλείται εξαιτίας του διαφορετικού τρόπου με τον οποίο το BIOS και το boot block αντιλαμβάνονται τους σκληρούς δίσκους. Το πρόβλημα συνήθως εμφανίζεται σε συστήματα με δύο δίσκους IDE, ειδικά όταν οι δίσκοι είναι master (ή μόνοι τους) ο καθένας στο δικό του ελεγκτή IDE και με το FreeBSD να είναι εγκαταστημένος στο δίσκο που βρίσκεται στο δευτερεύοντα ελεγκτή. Το boot block νομίζει ότι το σύστημα είναι εγκατεστημένο στον ad0 (το δεύτερο δίσκο του BIOS) ενώ ο πυρήνας αναθέτει τον πρώτο δίσκο στο δευτερεύοντα ελεγκτή, ad2. Μετά την ανίχνευση των συσκευών, ο πυρήνας προσπαθεί να προσαρτήσει αυτό που το boot block πιστεύει ότι είναι ο δίσκος εκκίνησης, ad0 ενώ στην πραγματικότητα είναι ο ad2 και φυσικά αποτυγχάνει.

Για να διορθώσετε το πρόβλημα, κάντε ένα από τα παρακάτω:

1. Επανεκκινήστε το σύστημα και πιέστε `Enter` στην προτροπή `Booting kernel in 10 seconds; hit [Enter] to interrupt`. Με τον τρόπο αυτό θα βγείτε στο πρόγραμμα του φορτωτή εκκίνησης.

   Κατόπιν γράψτε `set root_disk_unit="disk_number"`. Το *disk_number* θα είναι `0` αν το FreeBSD είναι εγκατεστημένο στο master δίσκο του πρώτου ελεγκτή IDE, `1` αν είναι εγκατεστημένο στο slave δίσκο του πρώτου ελεγκτή, `2` αν είναι εγκατεστημένος στον master δίσκο του δευτερεύοντος IDE καναλιού και τέλος, `3` αν είναι εγκατεστημένο στο slave δίσκο του δευτερεύοντος IDE καναλιού.

   Έπειτα γράψτε `boot`, και το σύστημα σας θα πρέπει να εκκινήσει κανονικά.

   Για να κάνετε μόνιμη αυτή την αλλαγή (ώστε να μην χρειάζεται να κάνετε το παραπάνω κάθε φορά που επανεκκινείτε ή ενεργοποιείτε το FreeBSD μηχάνημα σας), βάλτε τη γραμμή `root_disk_unit="disk_number "` στο αρχείο /boot/loader.conf.local.

2. Μετακινήστε το δίσκο του FreeBSD στον πρωτεύοντα ελεγκτή IDE, ώστε οι σκληροί δίσκοι να είναι συνεχόμενοι.

3.23. Ποια είναι τα όρια της μνήμης;

Το όριο είναι τα 4 gigabytes σε μια συνηθισμένη εγκατάσταση σε αρχιτεκτονική i386™. Ξεκινώντας από τις εκδόσεις FreeBSD 4.9 και 5.1, υποστηρίζεται και περισσότερη μνήμη μέσω του pae(4). Χρειάζεται ωστόσο να μεταγλωττιστεί ξανά ο πυρήνας περιλαμβάνοντας και μια έξτρα επιλογή για την ενεργοποίηση του PAE:

```
options       PAE
```

Το FreeBSD/pc98 έχει όριο τα 4 GB μνήμης, και δεν μπορεί να χρησιμοποιηθεί PAE σε αυτή την αρχιτεκτονική. Στο FreeBSD/alpha, το όριο της μνήμης εξαρτάται από τον τύπο του υλικού που χρησιμοποιείται - για λεπτομέρειες δείτε τις Σημειώσεις Έκδοσης Υλικού για Alpha. Άλλες αρχιτεκτονικές που υποστηρίζονται από το FreeBSD, έχουν αρκετά μεγαλύτερα θεωρητικά όρια

σχετικά με τη μέγιστη ποσότητα μνήμης (πολλά terabytes).

3.24. Ποια είναι τα όρια του συστήματος αρχείων ffs;

Για συστήματα αρχείων ffs, το μέγιστο θεωρητικό όριο είναι τα 8 terabytes (2G blocks), ή 16TB για προεπιλεγμένο μέγεθος block των 8K. Στην πραγματικότητα, υπάρχει ένα αρχικό όριο 1 terabyte, αλλά με κάποιες μετατροπές, είναι δυνατόν να δημιουργηθούν (και υπάρχουν) συστήματα αρχείων μεγέθους 4 terabytes.

Το μέγιστο μέγεθος ενός αρχείου σε ένα σύστημα ffs είναι περίπου 1G blocks, ή 4TB με μέγεθος block των 4K.

Πίνακας 1. Μέγιστα μεγέθη αρχείων

Μέγεθος block fs	**λειτουργεί**	**πρέπει να λειτουργεί**
4K	4T-1	4T
8K	32G	32T-1
16K	128G	32T-1
32K	512G	64T-1
64K	2048G	128T-1

Όταν το μέγεθος block του fs είναι 4K, λειτουργούν τα τριπλά έμμεσα blocks (triple indirect blocks) και τα πάντα θα έπρεπε να περιορίζονται μόνο από το μέγιστο αριθμό block που μπορεί να αναπαρασταθεί με τη χρήση τριπλών έμμεσων blocks (περίπου 1K^3 + 1K^2 + 1K), αλλά τελικά ο περιορισμός οφείλεται σε ένα (λάθος) όριο 1G-1 στους αριθμούς των blocks. Το όριο στους αριθμούς των block θα έπρεπε να είναι 2G-1. Υπάρχουν κάποια προβλήματα όταν οι αριθμοί των block του fs πλησιάζουν το 2G-1, αλλά τέτοιοι αριθμοί block δεν μπορούν να προσεγγιστούν όταν το μέγεθος block fs είναι 4K.

Για μεγέθη block 8K και μεγαλύτερα, τα πάντα θα έπρεπε να περιορίζονται από το όριο 2G-1 στους αριθμούς των block του fs, αλλά στην πραγματικότητα ο περιορισμός οφείλεται στο λάθος όριο 1G-1 στους αριθμούς των block του fs. Η χρήση του σωστού ορίου των 2G-1 blocks, δημιουργεί πράγματι προβλήματα.

3.25. Γιατί παίρνω το μήνυμα λάθος, archsw.readin.failed μετά την μεταγλώττιση και εκκίνηση νέου πυρήνα;

Γιατί ο πυρήνας σας και το υπόλοιπο του βασικού συστήματος (world) είναι εκτός συγχρονισμού. Η λειτουργία σε αυτή την κατάσταση δεν υποστηρίζεται. Βεβαιωθείτε ότι χρησιμοποιείτε τις εντολές `make buildworld` και `make buildkernel` για να αναβαθμίσετε τον πυρήνα σας.

Μπορείτε να εκκινήσετε ορίζοντας τον πυρήνα απευθείας από το δεύτερο στάδιο, πιέζοντας οποιοδήποτε πλήκτρο μόλις δείτε το | και πριν ξεκινήσει ο loader.

3.26. Η εγκατάσταση καταρρέει κατά την εκκίνηση. Τι μπορώ να κάνω;

Δοκιμάστε να απενεργοποιήσετε την υποστήριξη ACPI. Μόλις ξεκινήσει ο φορτωτής εκκίνησης, πιέστε το πλήκτρο space. Το σύστημα σας θα εμφανίσει

```
OK
```

1. Γράψτε

```
unset acpi_load
```

και κατόπιν

```
boot
```

Chapter 4. Συμβατότητα Υλικού

4.1. Γενικά

4.1.1. Θέλω να αγοράσω υλικό για το FreeBSD σύστημα μου. Ποιο μοντέλο / μάρκα / τύπος είναι το καλύτερο;

Υπάρχουν συνέχεια συζητήσεις για το θέμα αυτό στις λίστες ηλεκτρονικού ταχυδρομείου του FreeBSD. Αυτό ωστόσο είναι αναμενόμενο, καθώς το υλικό των υπολογιστών αλλάζει πολύ γρήγορα. Εμείς *εξακολουθούμε* να συνιστούμε με έμφαση, να διαβάσετε τις Σημειώσεις Υλικού του FreeBSD 12.0 ή 11.2 και να ψάξετε στα αρχεία των λιστών ηλεκτρονικού ταχυδρομείου, πριν αρχίσετε να ρωτάτε σχετικά με το τελευταίο και καλύτερο υλικό. Είναι αρκετά πιθανό να διαπιστώσετε, ότι για το υλικό που αναζητάτε, υπήρξε σχετική συζήτηση μόλις πριν μια εβδομάδα.

Αν ψάχνετε για φορητό υπολογιστή, ελέγξτε τα αρχεία τής λίστας ηλεκτρονικού ταχυδρομείου freebsd-mobile. Διαφορετικά, μάλλον θα θέλετε να δείτε τα αρχεία της freebsd-questions ή πιθανόν μια λίστα που να εξειδικεύεται στον τύπο του υλικού που ψάχνετε.

4.2. Αρχιτεκτονικές και Επεξεργαστές

4.2.1. Υποστηρίζει το FreeBSD αρχιτεκτονικές διαφορετικές από την x86;

Ναι. Αυτή τη στιγμή το FreeBSD μπορεί να εκτελεστεί σε αρχιτεκτονικές x86 και DEC (τώρα πλέον Compaq) Alpha. Από το FreeBSD 5.0 και μετά, υποστηρίζονται επίσης οι αρχιτεκτονικές AMD64 και Intel EM64T, η IA-64 καθώς και η sparc64. Αρχιτεκτονικές που θα υποστηρίζονται μελλοντικά, περιλαμβάνουν την MIPS® και PowerPC®. Μπορείτε να εγγραφείτε στις λίστες ταχυδρομείου freebsd-ppc και freebsd-mips αντίστοιχα για πληροφορίες σχετικά με την πρόοδο των εργασιών σε αυτές τις αρχιτεκτονικές. Για γενικές πληροφορίες σχετικά με νέες αρχιτεκτονικές, εγγραφείτε στην ηλεκτρονική λίστα του FreeBSD για τις μη-Intel πλατφόρμες.

Αν το μηχάνημα σας είναι διαφορετικής αρχιτεκτονικής και χρειάζεστε λειτουργικό άμεσα, σας συνιστούμε να ρίξετε μια ματιά στα NetBSD ή OpenBSD.

4.2.2. Υποστηρίζει το FreeBSD Συμμετρική Πολυεπεξεργασία (SMP);

Ναι. Το SMP ήταν ενεργοποιημένο από προεπιλογή στον πυρήνα *GENERIC* ήδη από το FreeBSD 5.2.

Η αρχική πρόθεση ήταν να είναι ενεργοποιημένο επίσης από προεπιλογή και στον πυρήνα τής έκδοσης 5.3 του FreeBSD, αλλά λόγω κάποιων προβλημάτων στην εκτέλεση του πυρήνα SMP σε μηχανήματα χωρίς πολλαπλούς επεξεργαστές, αποφασίστηκε να μείνει ανενεργό μέχρι την αντιμετώπιση τους. Αυτό ωστόσο αποτελεί προτεραιότητα για την έκδοση 5.4 του FreeBSD.

4.3. Σκληροί δίσκοι, μονάδες ταινίας, οδηγοί CD και DVD

4.3.1. Ποια είδη σκληρών δίσκων υποστηρίζονται από το FreeBSD;

Το FreeBSD υποστηρίζει μονάδες δίσκου των τύπων EIDE, SATA, SCSI, και SAS (με τον κατάλληλο συμβατό ελεγκτή - δείτε την επόμενη ενότητα), καθώς και όλους τους οδηγούς που χρησιμοποιούν το αρχικό interface της "Western Digital" (δηλ. MFM, RLL, ESDI και φυσικά IDE). Ίσως να μη λειτουργήσουν κάποιοι ελεγκτές ESDI που χρησιμοποιούν μη-τυποποιημένο interface. Καλύτερα να παραμείνετε σε interfaces τύπου WD1002/3/6/7 και αντίστοιχα τους.

4.3.2. Ποιοι ελεγκτές SCSI ή SAS υποστηρίζονται;

Δείτε την πλήρη λίστα στις Σημειώσεις Υλικού του FreeBSD 12.0 ή 11.2.

4.3.3. Τι τύποι οδηγών ταινίας υποστηρίζονται;

Το FreeBSD υποστηρίζει οδηγούς SCSI και QIC-36 (με interface QIC-02). Περιλαμβάνονται οδηγοί 8-mm (γνωστοί ως Exabyte) καθώς και οδηγοί DAT.

Ορισμένοι από τους πρώτους οδηγούς 8-mm δεν είναι ιδιαίτερα συμβατοί με SCSI-2, και μπορεί να μην λειτουργούν ικανοποιητικά με το FreeBSD.

4.3.4. Υποστηρίζει το FreeBSD μονάδες εναλλαγής ταινιών (tape changers);

Το FreeBSD υποστηρίζει μονάδες εναλλαγής τύπου SCSI με την χρήση της συσκευής ch(4) και της εντολής chio(1). Μπορείτε να βρείτε τις λεπτομέρειες σχετικά με τον τρόπο ελέγχου της μονάδας εναλλαγής στη σελίδα manual του chio(1).

Αν δεν χρησιμοποιείτε το AMANDA ή κάποιο άλλο προϊόν που να γνωρίζει πως να χειριστεί την μονάδα εναλλαγής ταινιών, θα πρέπει να θυμάστε ότι γενικά τα προγράμματα γνωρίζουν μόνο πως να κινήσουν μια ταινία από ένα σημείο σε ένα άλλο, και θα πρέπει εσείς να σημειώσετε σε ποια θέση (slot) βρίσκεται η ταινία, και σε ποια θέση πρέπει να πάει η ταινία που βρίσκεται αυτή τη στιγμή μέσα στον οδηγό.

4.4. Ποιοι οδηγοί CDROM υποστηρίζονται από το FreeBSD;

Υποστηρίζεται οποιοσδήποτε οδηγός SCSI που είναι συνδεμένος σε αντίστοιχα υποστηριζόμενο ελεγκτή.

Υποστηρίζονται ακόμα τα ακόλουθα μη-τυποποιημένα CDROM interfaces:

- Mitsumi LU002 (8bit), LU005 (16bit) and FX001D (16bit ταχύτητας 2x).
- Sony CDU 31/33A
- Sound Blaster μη-SCSI CDROM
- Matsushita/Panasonic CDROM
- IDE CDROM συμβατά με ATAPI

Όλες οι κάρτες που δεν είναι SCSI είναι γενικά εξαιρετικά πιο αργές όταν συγκρίνονται με

αντίστοιχους SCSI οδηγούς, και κάποιοι οδηγοί τύπου ATAPI ίσως να μη λειτουργήσουν.

Τα επίσημα FreeBSD CDROM ISO, καθώς και τα CDROM από το Daemon News και το FreeBSD Mall, υποστηρίζουν εκκίνηση απευθείας από το CD.

4.4.1. Ποιοι οδηγοί CD-RW υποστηρίζονται από το FreeBSD;

Το FreeBSD υποστηρίζει οποιοδήποτε οδηγό IDE CD-R ή CD-RW συμβατό με ATAPI. Δείτε το burncd(8) για λεπτομέρειες.

Το FreeBSD υποστηρίζει επίσης οποιοδήποτε οδηγό SCSI CD-R ή CD-RW. Εγκαταστήστε και χρησιμοποιήστε την εντολή `cdrecord` από την συλλογή των ports ή από πακέτο, και βεβαιωθείτε ότι έχετε ενσωματώσει στον πυρήνα σας τη συσκευή pass.

4.4.2. Υποστηρίζει το FreeBSD οδηγούς Zip®;

Το FreeBSD υποστηρίζει εγγενώς οδηγούς Zip® SCSI και ATAPI. Οι οδηγοί SCSI ZIP μπορούν να λειτουργήσουν μόνο αν έχουν ρυθμιστεί σε SCSI IDs 5 ή 6, αλλά αν το υποστηρίζει ο ελεγκτής SCSI που χρησιμοποιείτε, μπορείτε ακόμα και να εκκινήσετε από αυτούς. Δεν είναι ξεκάθαρο ποιοι ελεγκτές SCSI υποστηρίζουν εκκίνηση από συσκευές με ID διαφορετικό από 0 ή 1, έτσι θα πρέπει να συμβουλευθείτε την τεκμηρίωση του ελεγκτή σας, αν θέλετε να χρησιμοποιήσετε αυτή τη δυνατότητα.

Το FreeBSD υποστηρίζει επίσης οδηγούς Zip παράλληλης θύρας. Βεβαιωθείτε ότι ο πυρήνας σας περιέχει τα προγράμματα οδήγησης για τις συσκευές scbus0, da0, ppbus0, και vp0 (ο πυρήνας GENERIC περιέχει τα πάντα εκτός από το vp0). Με χρήση αυτών των προγραμμάτων οδήγησης, ο οδηγός της παράλληλης θύρας θα πρέπει να είναι διαθέσιμος ως συσκευή /dev/da0s4. Μπορείτε να προσαρτήσετε δίσκους χρησιμοποιώντας την εντολή `mount /dev/da0s4 /mnt` ή (για δίσκους που έχουν διαμορφωθεί μέσω dos) την `mount_msdos /dev/da0s4 /mnt`.

Διαβάστε επίσης το FAQ σχετικά με αφαιρούμενους οδηγούς σε επόμενο σημείο αυτού του κεφαλαίου, καθώς και τις σημειώσεις σχετικά με την "διαμόρφωση" στο κεφάλαιο Διαχείρισης.

4.4.3. Υποστηρίζει το FreeBSD Jaz®, EZ και άλλους αφαιρούμενους οδηγούς;

Απλώς λειτουργούν. Οι περισσότερες από τις συσκευές αυτές είναι SCSI, και έτσι δείχνουν σαν SCSI δίσκοι στο FreeBSD. Το IDE EZ εμφανίζεται σαν οδηγός IDE.

Βεβαιωθείτε ότι έχετε ενεργοποιήσει τυχόν εξωτερικές συσκευές πριν εκκινήσετε το σύστημα σας.

Για να αλλάξετε μέσο αποθήκευσης εν ώρα λειτουργίας, ελέγξτε τις mount(8), umount(8), και την camcontrol(8) (για συσκευές SCSI) ή atacontrol(8) (για συσκευές IDE), και επίσης τις συζητήσεις σχετικά με τη χρήση αφαιρούμενων οδηγών σε επόμενο τμήμα του FAQ.

4.5. Πληκτρολόγια και ποντίκια

4.5.1. Υποστηρίζει το FreeBSD το USB πληκτρολόγιο μου;

Το FreeBSD υποστηρίζει εγγενώς USB πληκτρολόγια. Ενεργοποιήστε την υποστήριξη USB στο

/etc/rc.conf.

Μόλις ενεργοποιηθεί η υποστήριξη USB πληκτρολογίου στο σύστημα σας, το πληκτρολόγιο τύπου AT αναγνωρίζεται ως /dev/kbd0 και το USB πληκτρολόγιο γίνεται /dev/kbd1, αν είναι και τα δύο συνδεμένα στο σύστημα. Αν υπάρχει μόνο το USB πληκτρολόγιο, θα αναγνωριστεί ως /dev/ukbd0.

Αν θέλετε να χρησιμοποιήσετε το USB πληκτρολόγιο στην κονσόλα, θα πρέπει να δηλώσετε συγκεκριμένα στον οδηγό της κονσόλας να χρησιμοποιήσει το υπάρχον USB πληκτρολόγιο. Αυτό μπορεί να γίνει εκτελώντας την ακόλουθη εντολή ως μέρος της διαδικασίας αρχικοποίησης του συστήματος:

```
# kbdcontrol -k /dev/kbd1 < /dev/ttyv0 > /dev/null
```

Παρατηρήστε ότι αν το πληκτρολόγιο USB είναι το μοναδικό πληκτρολόγιο, θα είναι διαθέσιμο ως /dev/ukbd0, και η εντολή θα δείχνει όπως παρακάτω:

```
# kbdcontrol -k /dev/ukbd0 < /dev/ttyv0 > /dev/null
```

Ένα καλό μέρος για να προσθέσετε την παραπάνω εντολή, είναι το αρχείο /etc/rc.i386.

Μόλις γίνει αυτό, το USB πληκτρολόγιο θα πρέπει να λειτουργεί και στο X περιβάλλον, χωρίς ανάγκη ειδικών ρυθμίσεων.

Η εν θερμώ σύνδεση και αποσύνδεση USB πληκτρολογίου, ίσως να μη λειτουργεί ακόμα σωστά. Σας συνιστούμε να συνδέσετε το πληκτρολόγιο πριν την εκκίνηση του συστήματος, και να το αφήσετε συνδεμένο μέχρι τον τερματισμό, για να αποφύγετε τυχόν προβλήματα.

Δείτε τη σελίδα manual ukbd:(4) για περισσότερες πληροφορίες.

4.5.2. Έχω ένα μη-τυπικό ποντίκι τύπου bus. Πως θα το ρυθμίσω;

Το FreeBSD υποστηρίζει ποντίκια τύπου bus και τον τύπο InPort bus από κατασκευαστές όπως Microsoft, Logitech και ATI. Ο πυρήνας GENERIC δεν περιέχει τον απαραίτητο οδηγό συσκευής. Για να περιλάβετε το πρόγραμμα οδήγησης στο δικό σας προσαρμοσμένο πυρήνα, προσθέστε την ακόλουθη γραμμή στο αρχείο ρυθμίσεων πυρήνα:

```
device mse0 at isa? port 0x23c irq5
```

Τα ποντίκια τύπου bus συνήθως έρχονται με δικές τους κάρτες επέκτασης. Ενδεχομένως να έχετε δυνατότητα να ρυθμίσετε την κάρτα σε διαφορετική διεύθυνση θύρας και IRQ από αυτά που φαίνονται παραπάνω. Συμβουλευθείτε το εγχειρίδιο του ποντικιού σας και τη σελίδα manual mse(4) για περισσότερες πληροφορίες.

4.6. Πως μπορώ να χρησιμοποιήσω το ποντίκι μου τύπου PS/2 (πόρτας ποντικιού ή πληκτρολογίου);

Το ποντίκι τύπου PS/2 υποστηρίζεται εγγενώς. Το απαραίτητο πρόγραμμα οδήγησης, psm, περιλαμβάνεται στον πυρήνα.

Αν ο προσαρμοσμένος πυρήνας σας δεν τον περιέχει, προσθέστε την ακόλουθη γραμμή στο αρχείο ρυθμίσεων πυρήνα, και μεταγλωττίστε ξανά τον πυρήνα σας.

```
device psm0 at atkbdc? irq 12
```

Μόλις ο πυρήνας ανιχνεύσει σωστά τη συσκευή psm0 κατά την εκκίνηση, βεβαιωθείτε ότι υπάρχει η αντίστοιχη καταχώρηση για το psm0 στον κατάλογο /dev. Μπορείτε να το δημιουργήσετε γράφοντας:

```
# cd /dev; sh MAKEDEV psm0
```

όταν έχετε εισέλθει ως χρήστης `root`.

Μπορείτε να παραλείψετε αυτό το βήμα αν χρησιμοποιείτε FreeBSD 5.0-RELEASE ή νεώτερο με ενεργοποιημένο το devfs(5), καθώς τα απαραίτητα αρχεία συσκευών θα δημιουργηθούν αυτόματα κάτω από τον κατάλογο /dev.

4.6.1. Είναι δυνατόν να χρησιμοποιηθεί το ποντίκι με κάποιο τρόπο έξω από το περιβάλλον του συστήματος X Window;

Αν χρησιμοποιείτε το προεπιλεγμένο πρόγραμμα οδήγησης κονσόλας, syscons(4), μπορείτε να χρησιμοποιήσετε το δείκτη του ποντικιού σας σε κονσόλες κειμένου για να κάνετε αποκοπή και επικόλληση κειμένου. Εκτελέστε τον δαίμονα του ποντικιού, moused(8), και ενεργοποιήστε το δείκτη του ποντικιού στην εικονική κονσόλα:

```
# moused -p /dev/xxxx -t yyyy
# vidcontrol -m on
```

Όπου το *xxxx* είναι το όνομα συσκευής του ποντικιού και το *yyyy* είναι ο τύπος του πρωτοκόλλου του. Ο δαίμονας του ποντικιού μπορεί να αναγνωρίσει αυτόματα το είδος του πρωτοκόλλου για τα περισσότερα ποντίκια, εκτός από παλιά σειριακά μοντέλα. Καθορίστε το πρωτόκολλο `auto` για να χρησιμοποιήσετε την αυτόματη ανίχνευση. Αν αυτή δεν δουλέψει, δείτε τη σελίδα manual moused(8) για μια λίστα με υποστηριζόμενους τύπους πρωτοκόλλων.

Αν έχετε ποντίκι τύπου PS/2, απλώς προσθέστε `moused_enable="YES"` στο αρχείο /etc/rc.conf για να εκκινεί ο δαίμονας του ποντικιού κατά την εκκίνηση. Επιπρόσθετα, αν επιθυμείτε να χρησιμοποιείτε το δαίμονα του ποντικιού σε όλες τις εικονικές κονσόλες, και όχι μόνο στην κονσόλα συστήματος, προσθέστε τη γραμμή `allscreens_flags="-m on"` στο /etc/rc.conf.

Όταν εκτελείται ο δαίμονας του ποντικιού, η πρόσβαση στο ποντίκι πρέπει να συντονίζεται μεταξύ του δαίμονα και άλλων προγραμμάτων, όπως τα X Windows. Κοιτάξτε στο FAQ την ερώτηση Γιατί το ποντίκι μου δεν δουλεύει στα X; για περισσότερες πληροφορίες σχετικά με αυτό το πρόβλημα.

4.6.2. Πως μπορώ να κάνω αποκοπή και επικόλληση κειμένου με το ποντίκι σε μια κονσόλα κειμένου;

Μόλις ενεργοποιήσετε το δαίμονα του ποντικιού (δείτε την προηγούμενη ενότητα), κρατήστε πιεσμένο το πλήκτρο 1 (το αριστερό πλήκτρο) και κινήστε το ποντίκι για να επιλέξετε μια περιοχή κειμένου. Κατόπιν, πιέστε το πλήκτρο 2 (το μεσαίο πλήκτρο) για να το επικολλήσετε στην περιοχή του δρομέα. Με την πίεση του πλήκτρου 3 (δεξιού πλήκτρου) μπορείτε να "επεκτείνετε" την επιλεγμένη περιοχή κειμένου.

Αν το ποντίκι σας δεν έχει μεσαίο πλήκτρο, μπορεί να θέλετε να το εξομοιώσετε ή να αλλάξετε τις λειτουργίες των πλήκτρων χρησιμοποιώντας τις επιλογές που παρέχονται από τον δαίμονα του ποντικιού Δείτε τη σελίδα manual moused(8) για τις λεπτομέρειες.

4.6.3. Το ποντίκι μου έχει διάφορα έξυπνα πλήκτρα και ροδέλα κύλισης. Μπορώ να τα χρησιμοποιήσω στο FreeBSD;

Η απάντηση, δυστυχώς, είναι "εξαρτάται". Τα ποντίκια με έξτρα δυνατότητες συνήθως απαιτούν εξειδικευμένα προγράμματα οδήγησης. Αν το πρόγραμμα οδήγησης του ποντικιού ή το αντίστοιχο πρόγραμμα του χρήστη δεν παρέχουν συγκεκριμένη υποστήριξη για το ποντίκι, θα λειτουργεί ως ένα απλό ποντίκι δύο ή τριών πλήκτρων.

Για πιθανή χρήση της ροδέλας σε περιβάλλον X Window, δείτε την αντίστοιχη ενότητα.

4.6.4. Πως μπορώ να χρησιμοποιήσω το ποντίκι / trackball / touchpad στον φορητό μου υπολογιστή;

Δείτε την απάντηση στην προηγούμενη ερώτηση.

4.6.5. Πως μπορώ να χρησιμοποιήσω το πλήκτρο delete στο sh και csh;

Για το Κέλυφος Bourne, προσθέστε τις ακόλουθες γραμμές στο αρχείο σας .shrc. Δείτε επίσης τις σελίδες manual sh(1) και editrc(5).

```
bind ^? ed-delete-next-char # for console
bind ^[[3~ ed-delete-next-char # for xterm
```

Για το Κέλυφος C, προσθέστε τις ακόλουθες γραμμές στο αρχείο σας .cshrc. Δείτε επίσης τη σελίδα manual του csh(1).

```
bindkey ^? delete-char # for console
bindkey ^[[3~ delete-char # for xterm
```

Για περισσότερες πληροφορίες, δείτε αυτή τη σελίδα.

4.7. Συσκευές σειριακής επικοινωνίας και Δικτύωσης

4.7.1. Ποιες κάρτες δικτύου υποστηρίζει το FreeBSD;

Για την πλήρη λίστα, δείτε τις Σημειώσεις Υλικού που παρέχονται με κάθε έκδοση του FreeBSD.

4.7.2. Υποστηρίζει το FreeBSD modems που λειτουργούν με τη βοήθεια λογισμικού όπως τα Winmodems;

Το FreeBSD υποστηρίζει αρκετά software modems με την βοήθεια επιπρόσθετου λογισμικού. Το port comms/ltmdm προσθέτει υποστήριξη για modems που βασίζονται στο δημοφιλές κύκλωμα Lucent LT. Το port comms/mwavem υποστηρίζει το modem που διαθέτουν οι φορητοί υπολογιστές Thinkpad 600 και 700 της IBM.

Δεν μπορείτε να εγκαταστήσετε το FreeBSD μέσω software modem. Το λογισμικό αυτό πρέπει να εγκατασταθεί μετά την εγκατάσταση του FreeBSD.

4.7.3. Υπάρχει εγγενές πρόγραμμα οδήγησης για τις κάρτες Broadcom 43xx;

Όχι, και μάλλον δεν θα υπάρξει.

Η Broadcom αρνείται να δώσει δημόσια πληροφορίες σχετικά με τον προγραμματισμό των ολοκληρωμένων της που χρησιμοποιούνται σε εφαρμογές ασύρματων δικτύων, πιθανόν επειδή και το τμήμα του πομποδέκτη της κάρτας ελέγχεται με τη βοήθεια λογισμικού. Για να αποκτήσουν έγκριση για τις κάρτες τους από το FCC, πρέπει να εξασφαλίσουν ότι οι τελικοί χρήστες δεν θα είναι σε θέση να κάνουν ρυθμίσεις όπως αλλαγή της συχνότητας λειτουργίας, των παραμέτρων διαμόρφωσης και της ισχύος εκπομπής. Αλλά χωρίς τις πληροφορίες προγραμματισμού, είναι σχεδόν αδύνατο να γραφεί πρόγραμμα οδήγησης.

4.7.4. Ποιες κάρτες πολλαπλών σειριακών θυρών υποστηρίζονται από το FreeBSD;

Υπάρχει μια λίστα για αυτές στην ενότητα διάφορων συσκευών του Εγχειριδίου.

Ακόμα φαίνεται ότι λειτουργούν και κάποιες κάρτες που είναι αντιγραφές επώνυμων μοντέλων, ειδικά όσες υποστηρίζουν ότι είναι συμβατές με τις αντίστοιχες τις AST.

Δείτε τη σελίδα manual sio(4) για περισσότερες πληροφορίες σχετικά με τη ρύθμιση τέτοιων καρτών.

4.7.5. Πως μπορώ να εμφανίσω την προτροπή boot: σε μια σειριακή κονσόλα;

1. Δημιουργήστε πυρήνα που να περιέχει την επιλογή `options COMCONSOLE`.
2. Δημιουργήστε το /boot.config και γράψτε μέσα σε αυτό μόνο την επιλογή `-P`.
3. Αποσυνδέστε το πληκτρολόγιο από το σύστημα.

Δείτε το αρχείο /usr/src/sys/i386/boot/biosboot/README.serial για περισσότερες πληροφορίες.

4.8. Συσκευές ήχου

4.8.1. Ποιες κάρτες ήχου υποστηρίζονται από το FreeBSD;

Το FreeBSD υποστηρίζει διάφορες κάρτες ήχου, συμπεριλαμβανομένων των SoundBlaster®, SoundBlaster® Pro, SoundBlaster® 16, Pro Audio Spectrum 16, AdLib, και Gravis UltraSound (για περισσότερες πληροφορίες, δείτε τις Πληροφορίες Έκδοσης του FreeBSD και τη σελίδα manual snd(4)). Υπάρχει επίσης περιορισμένη υποστήριξη για κάρτες MIDI που είναι συμβατές με το πρότυπο MPU-401. Επίσης υποστηρίζονται οι κάρτες που είναι συμβατές με το πρότυπο Microsoft® Sound System.

Το παραπάνω ισχύει μόνο για τον ήχο! Το πρόγραμμα οδήγησης δεν υποστηρίζει τυχόν CDROM, SCSI ή joysticks που συνδέονται πάνω σε αυτές τις κάρτες, εκτός από την SoundBlaster®. Αν και η διεπαφή SCSI της SoundBlaster® καθώς και κάποια μη-SCSI CDROM υποστηρίζονται, δεν μπορούν ωστόσο να χρησιμοποιηθούν για τη διαδικασία εκκίνησης.

4.8.2. Υπάρχει κάποια λύση για το πρόβλημα του ήχου στην κάρτα μου που υποστηρίζεται από το pcm(4);

Κάποιες κάρτες ήχου, όπως η es1370, μηδενίζουν την ένταση του ήχου σε κάθε εκκίνηση. Πρέπει να εκτελείτε την ακόλουθη εντολή κάθε φορά που ξεκινά το μηχάνημα:

```
# mixer pcm 100 vol 100 cd 100
```

4.9. Άλλο υλικό

4.9.1. Ποιες άλλες συσκευές υποστηρίζονται από το FreeBSD;

Δείτε το Εγχειρίδιο για τη λίστα των υπόλοιπων συσκευών που υποστηρίζονται.

4.9.2. Υποστηρίζει το FreeBSD διαχείριση ενέργειας για το φορητό μου υπολογιστή;

Από το FreeBSD 4.X και μετά, υποστηρίζεται το APM σε συγκεκριμένα μηχανήματα. Περισσότερες πληροφορίες μπορείτε να βρείτε στο apm(4).

Από το FreeBSD 5.X και μετά, υποστηρίζεται η δυνατότητα ACPI η οποία υπάρχει σε όλους τους σύγχρονους υπολογιστές. Μπορείτε να βρείτε περισσότερες πληροφορίες στο acpi(4). Αν ένα σύστημα υποστηρίζει τόσο APM όσο και ACPI, μπορείτε να χρησιμοποιήσετε όποιο θέλετε. Σας συνιστούμε να δοκιμάσετε και τα δύο και να επιλέξετε αυτό που καλύπτει καλύτερα τις ανάγκες σας.

4.10. Πως μπορώ να απενεργοποιήσω το ACPI;

Προσθέστε τη γραμμή

```
hint.acpi.0.disabled="1"
```

στο αρχείο /boot/device.hints.

4.10.1. Γιατί το Micron σύστημα μου κρεμάει κατά την εκκίνηση;

Ορισμένες μητρικές Micron υλοποιούν το PCI BIOS με μη τυποποιημένο τρόπο, προκαλώντας προβλήματα στην εκκίνηση του FreeBSD, καθώς οι PCI συσκευές δεν ρυθμίζονται στις διευθύνσεις που αναφέρονται.

Για να παρακάμψετε το πρόβλημα, απενεργοποιήστε την επιλογή "Plug and Play Operating System" από το BIOS.

4.10.2. Η δισκέτα εκκίνησης κρεμάει στη μητρική ASUS K7V. Πως μπορώ να το διορθώσω αυτό;

Στις ρυθμίσεις του BIOS, απενεργοποιήστε την επιλογή "boot virus protection".

4.10.3. Γιατί η PCI κάρτα δικτύου μου της 3Com® δεν λειτουργεί με το Micron υπολογιστή μου;

Ορισμένες μητρικές της Micron έχουν μη-τυποποιημένο PCI BIOS το οποίο δεν ρυθμίζει τις συσκευές PCI στις διευθύνσεις που αναφέρονται. Αυτό δημιουργεί προβλήματα κατά την εκκίνηση του FreeBSD.

Για να παρακάμψετε το πρόβλημα, απενεργοποιήστε την επιλογή "Plug and Play Operating System" από το BIOS.

4.10.4. Η PCMCIA κάρτα μου δεν λειτουργεί. Βλέπω το εξής μήνυμα: cbb0: unsupported card type detected. Τι μπορώ να κάνω;

Μπορείτε να δοκιμάσετε να χρησιμοποιήσετε την αρχική υλοποίηση OLDCARD. Τροποποιήστε το αρχείο ρύθμισης του πυρήνα σας, και αφαιρέστε τις ακόλουθες γραμμές:

```
device cbb
device pccard
device cardbus
```

Μετά προσθέστε:

```
device pcic
```

```
device card 1
```

Μεταγλωττίστε ξανά και εγκαταστήστε το νέο πυρήνα όπως περιγράφεται στην Ρύθμιση Πυρήνα του FreeBSD.

Chapter 5. Αντιμετώπιση Προβλημάτων

5.1. Γιατί το FreeBSD βρίσκει λάθος ποσότητα μνήμης;

Αυτό οφείλεται στη διαφορά μεταξύ φυσικών και εικονικών διευθύνσεων μνήμης.

Η σύμβαση που κατά βάση ακολουθείται στο υλικό του PC, είναι να χρησιμοποιείται η μνήμη μεταξύ 3.5G και 4G για ειδικό σκοπό, συνήθως για την πρόσβαση σε κάρτες PCI. Αυτό έχει ως αποτέλεσμα να μην μπορεί να αντιστοιχηθεί φυσική μνήμη σε αυτή την περιοχή διευθύνσεων.

Το υλικό του υπολογιστή σας θα καθορίσει τι γίνεται με την μνήμη που κανονικά εμφανίζεται σε αυτή τη θέση. Δυστυχώς, σε κάποιες περιπτώσεις το υλικό δεν κάνει τίποτα, και χάνεται η δυνατότητα χρήσης των τελευταίων 500M μνήμης RAM.

Ευτυχώς, στις περισσότερες περιπτώσεις το υλικό ανακατευθύνει τη μνήμη σε υψηλότερη θέση, ώστε να είναι ακόμα δυνατή η χρήση της. Αυτό μπορεί ωστόσο να σας προκαλέσει κάποια σύγχυση αν παρακολουθείτε τα μηνύματα εκκίνησης.

Στην 32 bit έκδοση του FreeBSD, η μνήμη φαίνεται να έχει χαθεί καθώς ανακατευθύνεται πάνω από τα 4G, τα οποία δεν είναι προσβάσιμα από 32 bit πυρήνα. Στην περίπτωση αυτή η λύση είναι να φτιάξετε ένα πυρήνα τύπου PAE. Δείτε αυτήν την καταχώρηση στο FAQ για περισσότερες πληροφορίες.

Στην 64 bit έκδοση του FreeBSD, ή όταν χρησιμοποιείται πυρήνας τύπου PAE, το FreeBSD θα ανιχνεύσει και θα ανακατευθύνει σωστά τη μνήμη ώστε να είναι χρησιμοποιήσιμη. Κατά την εκκίνηση ωστόσο, μπορεί να φαίνεται ότι το FreeBSD ανιχνεύει περισσότερη μνήμη από αυτή που έχει στην πραγματικότητα το σύστημα. Αυτό είναι φυσιολογικό και η διαθέσιμη μνήμη θα διορθωθεί καθώς ολοκληρώνεται η διαδικασία της εκκίνησης.

5.2. Ο σκληρός μου δίσκος έχει χαλασμένους τομείς. Τι μπορώ να κάνω;

Στους δίσκους SCSI, ο οδηγός μπορεί συνήθως να επανατοποθετήσει αυτόματα τα δεδομένα σε εναλλακτικούς τομείς. Ωστόσο οι περισσότεροι δίσκοι έρχονται με την δυνατότητα αυτή απενεργοποιημένη.

Για να ενεργοποιήσετε την επανατοποθέτηση χαλασμένων τομέων, επεξεργαστείτε την πρώτη σελίδα κατάστασης της συσκευής (modepage), δίνοντας την παρακάτω εντολή (ως `root`):

```
# camcontrol modepage sd0 -m 1 -e -P 3
```

και αλλάξτε τις τιμές των AWRE και ARRE από 0 σε 1:

```
AWRE (Auto Write Reallocation Enbld):  1
ARRE (Auto Read Reallocation Enbld):  1
```

Οι σύγχρονοι οδηγοί τύπου IDE έχουν επίσης ενεργοποιημένη από το εργοστάσιο τη δυνατότητα επανατοποθέτησης χαλασμένων τομέων.

Αν δείτε προειδοποιήσεις σχετικά με χαλασμένους τομείς (σε οποιοδήποτε είδος δίσκου), είναι ώρα να σκεφτείτε να αλλάξετε τον οδηγό. Ίσως μπορέσετε να χρησιμοποιήσετε το διαγνωστικό πρόγραμμα που δίνει ο κατασκευαστής του δίσκου για να απομονώσετε τους χαλασμένους τομείς, αλλά στην καλύτερη περίπτωση απλώς θα κερδίσετε λίγο περισσότερο χρόνο.

5.3. Γιατι το FreeBSD δεν ανιχνεύει τον ελεγκτή SCSI στον HP Netserver;

Το πρόβλημα αυτό είναι γνωστό. Ο ενσωματωμένος στη μητρική ελεγκτής SCSI του HP Netserver, χρησιμοποιεί σύνδεση τύπου EISA και καταλαμβάνει τη θέση EISA με αριθμό 11. Με τον τρόπο αυτό, όλες οι "πραγματικές" υποδοχές τύπου EISA βρίσκονται πριν από αυτή. Ωστόσο, η περιοχή διευθύνσεων των υποδοχών EISA με αριθμό = 10, συγκρούεται με την περιοχή διευθύνσεων του PCI, και το FreeBSD στη σημερινή του μορφή, δεν μπορεί να χειριστεί σωστά αυτή την κατάσταση.

Έτσι, για την ώρα, το καλύτερο που μπορείτε να κάνετε είναι να παριστάνετε ότι δεν υπάρχει σύγκρουση διευθύνσεων :) και να ανεβάσετε την επιλογή `EISA_SLOTS` του πυρήνα στην τιμή 12. Μεταγλωττίστε έπειτα ξανά τον πυρήνα, όπως περιγράφεται στην σχετική καταχώρηση του Εγχειριδίου.

Φυσικά αυτό είναι ένα πρόβλημα αντίστοιχο με το αυγό και την κότα, όσο αφορά την εγκατάσταση ενός τέτοιου μηχανήματος. Για να προσπεράσετε το πρόβλημα, υπάρχει ειδική πρόβλεψη στο *UserConfig*. Μη χρησιμοποιήσετε το "visual" interface, αλλά την γραμμή εντολών. Απλώς γράψτε:

```
eisa 12
quit
```

στην προτροπή, και εγκαταστήστε το σύστημα σας όπως συνήθως. Σας συνιστούμε ωστόσο να μεταγλωττίσετε και να εγκαταστήσετε το δικό σας προσαρμοσμένο πυρήνα.

Ευελπιστούμε ότι σε μελλοντικές εκδόσεις, θα υπάρχει καλύτερη διόρθωση για το πρόβλημα αυτό.

> Δεν μπορείτε να χρησιμοποιήσετε δίσκο σε κατάσταση `dangerously dedicated (επικίνδυνα αφοσιωμένη)` με τον HP Netserver. Δείτε αυτή τη σημείωση για περισσότερες πληροφορίες.

5.4. Βλέπω συνέχεια μηνύματα του τύπου ed1: timeout. Τι σημαίνουν;

Τα μηνύματα αυτά προκαλούνται συνήθως από διενέξεις στα interrupts (π.χ. δύο κάρτες που χρησιμοποιούν το ίδιο IRQ). Εκκινήστε με την επιλογή -c και αλλάξτε την καταχώρηση ed0/de0/... ώστε να συμβαδίζει με το υλικό σας.

Αν χρησιμοποιείτε την σύνδεση BNC της κάρτας δικτύου σας, ίσως να δείτε επίσης αντίστοιχα

μηνύματα σε περίπτωση προβληματικού τερματισμού. Για να ελέγξετε την περίπτωση αυτή, συνδέστε ένα τερματιστή απευθείας στην κάρτα (χωρίς καλώδιο) και δείτε αν σταματήσουν τα μηνύματα.

Κάποιες κάρτες συμβατές με NE2000, δίνουν αυτό το μήνυμα αν δεν υπάρχει σύνδεση στη θύρα UTP ή αν το καλώδιο είναι αποσυνδεμένο.

5.5. Γιατί σταμάτησε να λειτουργεί η κάρτα μου 3Com® 3C509 χωρίς να υπάρχει εμφανής λόγος;

Η κάρτα αυτή έχει την κακή συνήθεια να χάνει τις ρυθμίσεις της. Ανανεώστε τις, χρησιμοποιώντας το βοηθητικό πρόγραμμα DOS `3c5x9.exe`.

5.6. Ο εκτυπωτής μου στην παράλληλη θύρα είναι απελπιστικά αργός. Τι μπορώ να κάνω;

Αν το μόνο πρόβλημα είναι ο υπερβολικά αργός εκτυπωτής, μπορείτε να δοκιμάσετε να αλλάξετε την κατάσταση λειτουργίας της παράλληλης θύρας όπως περιγράφεται στο κεφάλαιο του Εγχειριδίου σχετικά με την Εγκατάσταση Εκτυπωτή.

5.7. Γιατί τα προγράμματα μου περιστασιακά τερματίζουν με σφάλμα Signal 11;

Τα σφάλματα τύπου Signal 11 δημιουργούνται όταν μια διεργασία προσπαθεί να προσπελάσει περιοχή μνήμης για την οποία δεν έχει πάρει άδεια από το λειτουργικό σύστημα. Αν συμβαίνει κάτι τέτοιο σε φαινομενικά τυχαία χρονικά διαστήματα, θα πρέπει να αρχίσετε να το ερευνάτε πολύ προσεκτικά.

Τα προβλήματα αυτά συνήθως οφείλονται σε κάποιον από τους παρακάτω λόγους:

1. Αν το πρόβλημα εμφανίζεται μόνο σε μια συγκεκριμένη εφαρμογή την οποία αναπτύσσετε εσείς, είναι πιθανώς λάθος στον δικό σας κώδικα.
2. Αν το πρόβλημα βρίσκεται σε τμήμα του βασικού συστήματος του FreeBSD, μπορεί επίσης να είναι προβληματικός κώδικας, αλλά τις περισσότερες φορές, τα προβλήματα αυτά βρίσκονται και διορθώνονται πριν διανεμηθούν στους περισσότερους από εσάς που διαβάζετε το FAQ (για το λόγο αυτό άλλωστε υπάρχει και η γραμμή ανάπτυξης -current).

Για παράδειγμα, ένας γρήγορος τρόπος να διαπιστώσετε ότι *δεν* πρόκειται για πρόβλημα του FreeBSD, είναι αν το πρόβλημα εμφανίζεται κατά τη μεταγλώττιση κάποιου προγράμματος, αλλά κάθε φορά και σε διαφορετικό σημείο.

Για παράδειγμα, υποθέστε ότι εκτελείτε ένα "make buildworld", και η μεταγλώττιση αποτυγχάνει κατά την επεξεργασία του αρχείου ls.c σε ls.o. Αν εκτελέσετε ξανά "make buildworld", και η μεταγλώττιση σταματήσει στο ίδιο σημείο, πρόκειται πράγματι για πρόβλημα στα αρχεία του build — δοκιμάστε να ανανεώσετε τον πηγαίο κώδικα και να ξαναπροσπαθήσετε. Αν η μεταγλώττιση αποτυγχάνει αλλού, αυτό σχεδόν σίγουρα οφείλεται σε προβληματικό υλικό.

Τι πρέπει να κάνετε:

Στην πρώτη περίπτωση μπορείτε να χρησιμοποιήσετε κάποιο debugger όπως το gdb για να βρείτε το σημείο στο πρόγραμμα με την προβληματική διεύθυνση και να το διορθώσετε.

Στη δεύτερη περίπτωση, θα πρέπει να επαληθεύσετε ότι δεν φταίει το υλικό σας.

Στις συνηθισμένες αιτίες αυτού του προβλήματος, περιλαμβάνονται:

1. Οι σκληροί σας δίσκοι μπορεί να υπερθερμαίνονται. Ελέγξτε ότι λειτουργούν οι ανεμιστήρες στο κουτί σας. Αν δεν λειτουργούν, είναι πιθανό οι δίσκοι σας (και ίσως και άλλα εξαρτήματα) να υπερθερμαίνονται.
2. Ο επεξεργαστής σας έχει υπερθερμανθεί: Αυτό μπορεί να συμβεί σε περίπτωση που τον λειτουργείτε σε μεγαλύτερη συχνότητα από την κανονική (overclocking) ή αν το ανεμιστηράκι του επεξεργαστή έχει σταματήσει να λειτουργεί. Σε κάθε περίπτωση, θα πρέπει να εξασφαλίσετε ότι χρησιμοποιείτε το υλικό σας σύμφωνα με τις προδιαγραφές του, τουλάχιστον για όσο διάστημα χρειάζεται για να επιλύσετε το πρόβλημα. Για παράδειγμα, αν έχετε κάνει overclocking, επιστρέψτε τον επεξεργαστή στην κανονική του συχνότητα.

   Σχετικά με το overclocking, σημειώστε επίσης ότι είναι φτηνότερο να έχετε ένα πιο αργό σύστημα από ένα κατεστραμμένο που χρειάζεται αντικατάσταση! Επίσης η κοινότητα γενικά δεν θα σας αντιμετωπίσει με κατανόηση αν αναφέρετε προβλήματα που παρουσιάζονται σε συστήματα που λειτουργούν εκτός προδιαγραφών, είτε εσείς πιστεύετε ότι η λειτουργία τους είναι ασφαλής, είτε όχι.

3. Προβληματική μνήμη: Αν έχετε εγκατεστημένα περισσότερα από ένα SIMMS / DIMMS, αφαιρέστε τα και προσπαθήστε να λειτουργήσετε το μηχάνημα με ένα-ένα χωριστά ώστε να εντοπίσετε το πρόβλημα σε επίπεδο ενός SIMM / DIMM, ή ίσως σε ένα συνδυασμό τους.
4. Υπερ-αισιόδοξες ρυθμίσεις μητρικής: Στις ρυθμίσεις του BIOS, και σε κάποιες περιπτώσεις σε ρυθμίσεις στη μητρική μέσω βραχυκυκλωτήρων (jumpers), υπάρχει η δυνατότητα μεταβολής διάφορων χρονισμών. Στις περισσότερες περιπτώσεις οι προεπιλεγμένες ρυθμίσεις είναι επαρκείς, και ίσως δημιουργήσετε προβλήματα αν ρυθμίσετε πολύ χαμηλά τις καταστάσεις αναμονής (wait states) της RAM ή θέσετε στο BIOS την επιλογή "RAM Speed: Turbo". Μια καλή ιδέα είναι να επιστρέψετε τις ρυθμίσεις του BIOS στις προεπιλεγμένες, αλλά πριν το κάνετε, σημειώστε κάπου τις δικές σας.
5. Ανεπαρκής ή κακής ποιότητας τροφοδοσία στη μητρική. Αν έχετε κάρτες I/O, σκληρούς δίσκους ή CDROM στο σύστημα σας που δεν χρησιμοποιείτε, δοκιμάστε να τα αφαιρέσετε ή να αποσυνδέσετε προσωρινά την παροχή τροφοδοσίας τους, για να διαπιστώσετε αν το τροφοδοτικό σας μπορεί να διαχειριστεί μικρότερο φορτίο. Ή απλώς δοκιμάστε ένα άλλο τροφοδοτικό, κατά προτίμηση ένα με λίγο μεγαλύτερη ισχύ (για παράδειγμα αν το τρέχον σας τροφοδοτικό είναι ονομαστικής ισχύος 250W, δοκιμάστε ένα ισχύος 300W).

Θα πρέπει επίσης να διαβάσετε το SIG11 FAQ (το οποίο φαίνεται παρακάτω) το οποίο περιλαμβάνει εξαιρετικές επεξηγήσεις για όλα αυτά τα προβλήματα, αν και πολλές από αυτές είναι γραμμένες από την σκοπιά του Linux®. Ένα ενδιαφέρον τμήμα του SIG11 FAQ είναι και αυτό που αναφέρεται στην πιθανότητα να μην ανιχνεύεται προβληματική μνήμη από διαγνωστικά προγράμματα ή συσκευές ελέγχου.

Τέλος, αν τίποτα από τα παραπάνω δεν βοηθήσει, είναι πιθανόν να έχετε εντοπίσει ένα πρόβλημα (bug) στο FreeBSD και θα πρέπει να ακολουθήσετε τις οδηγίες για να στείλετε αναφορά προβλήματος.

Μπορείτε να βρείτε εκτεταμένη ανάλυση στο FAQ σχετικά με το πρόβλημα SIG11.

5.8. Το σύστημα μου σταματάει είτε με Fatal trap 12: page fault in kernel mode, ή με panic:, δείχνοντας και μια σειρά από πληροφορίες. Τι πρέπει να κάνω;

Η ομάδα ανάπτυξης του FreeBSD ενδιαφέρεται ιδιαίτερα για αυτά τα λάθη, αλλά χρειάζεται περισσότερες πληροφορίες εκτός από το μήνυμα λάθους που βλέπετε. Αντιγράψτε το πλήρες μήνυμα και έπειτα συμβουλευθείτε την ενότητα του FAQ σχετικά με τα kernel panics, δημιουργήστε ένα πυρήνα με δυνατότητα εκσφαλμάτωσης (debugging kernel) και εκτελέστε ένα backtrace. Αυτό μπορεί να ακούγεται δύσκολο, αλλά δεν χρειάζεστε στην πραγματικότητα γνώσεις προγραμματισμού. Αρκεί να ακολουθήσετε τις οδηγίες.

5.9. Γιατί η οθόνη μου μαυρίζει και χάνει το συγχρονισμό της κατά την εκκίνηση;

Πρόκειται για γνωστό πρόβλημα με την κάρτα γραφικών ATI Mach64. Το πρόβλημα είναι ότι η κάρτα αυτή χρησιμοποιεί την διεύθυνση `2e8`, η οποία χρησιμοποιείται επίσης και από την τέταρτη σειριακή θύρα. Λόγω κάποιου προβλήματος (ή της σχεδίασης) του προγράμματος οδήγησης sio(4), το πρόγραμμα όχι μόνο θα προσπαθήσει να ανιχνεύσει αυτή τη διεύθυνση ακόμα και αν δεν έχετε τέταρτη σειριακή θύρα, αλλά *ακόμα* και στην περίπτωση που έχετε απενεργοποιήσει τη σειριακή θύρα sio3 (δηλ. την τέταρτη) η οποία φυσιολογικά χρησιμοποιεί αυτή τη διεύθυνση.

Μέχρι να διορθωθεί το πρόβλημα αυτό, μπορείτε να χρησιμοποιήσετε το παρακάτω τέχνασμα για να το παρακάμψετε:

1. Γράψτε `-c` στην προτροπή εκκίνησης. (Με τον τρόπο αυτό θα βάλετε τον πυρήνα σε κατάσταση ρύθμισης).
2. Απενεργοποιήστε τις sio0, sio1, sio2 και sio3 (όλες). Με τον τρόπο αυτό το πρόγραμμα οδήγησης δεν ενεργοποιείται καν, άρα δεν δημιουργείται πρόβλημα.
3. Γράψτε exit για να συνεχίσετε την εκκίνηση.

Αν θέλετε να χρησιμοποιήσετε τις σειριακές θύρες, θα πρέπει να δημιουργήσετε νέο πυρήνα, με την ακόλουθη μετατροπή: Στο αρχείο /usr/src/sys/i386/isa/sio.c βρείτε το πρώτο σημείο που εμφανίζεται το αλφαριθμητικό `0x2e8` και αφαιρέστε αυτό το αλφαριθμητικό και το κόμμα που βρίσκεται πριν από αυτό (κρατήστε το κόμμα που βρίσκεται μετά). Ακολουθήστε τώρα τη συνηθισμένη διαδικασία δημιουργίας νέου πυρήνα.

Ακόμα και μετά την εφαρμογή αυτών των διορθώσεων, ίσως ανακαλύψετε ότι το σύστημα X Window δεν λειτουργεί σωστά. Αν συμβαίνει αυτό, βεβαιωθείτε ότι χρησιμοποιείτε έκδοση 3.3.3 ή μεγαλύτερη του XFree86™. Από την έκδοση αυτή και μετά, υπάρχει ενσωματωμένη υποστήριξη για κάρτες Mach64 και επίσης διατίθεται εξειδικευμένος εξυπηρετητής X για την κάρτα αυτή.

5.10. Γιατί το FreeBSD σύστημα μου χρησιμοποιεί μόνο 64MB RAM, ενώ ο υπολογιστής μου έχει εγκατεστημένα 128MB;

Εξαιτίας του τρόπου με τον οποίο το FreeBSD διαβάζει το μέγεθος της μνήμης από το BIOS, μπορεί να ανιχνεύσει μόνο 16 bits μέγεθος σε Kbytes (65536 Kbytes = 64MB) (ή και λιγότερο... ορισμένα BIOS δίνουν προκαθορισμένο μέγεθος μνήμης 16M). Αν έχετε περισσότερα από 64MB, το FreeBSD θα προσπαθήσει να τα ανιχνεύσει. Η ανίχνευση ωστόσο μπορεί να αποτύχει.

Για να παρακάμψετε το πρόβλημα, θα πρέπει να χρησιμοποιήσετε την επιλογή του πυρήνα που φαίνεται παρακάτω. Υπάρχει τρόπος να ληφθούν πλήρεις πληροφορίες σχετικά με τη μνήμη από το BIOS, αλλά στο bootblock δεν υπάρχει αρκετός χώρος για να γίνει αυτό. Κάποια μέρα, όταν διορθωθεί το πρόβλημα της έλλειψης χώρου στα bootblocks, θα χρησιμοποιήσουμε τις εκτεταμένες λειτουργίες του BIOS για να ανακτήσουμε πλήρεις πληροφορίες σχετικά με τη μνήμη. Για την ώρα, πρέπει να περιοριστούμε στην ρύθμιση της αντίστοιχης επιλογής του πυρήνα.

```
options "MAXMEM=n"
```

Όπου το *n* είναι το μέγεθος της μνήμης σε kilobytes. Για μηχάνημα με 128 MB, θα πρέπει να χρησιμοποιήσετε το `131072`.

5.11. Το σύστημα μου έχει περισσότερο από 1 GB RAM, και παίρνω panics με μηνύματα kmem_map too small. Που είναι το πρόβλημα;

Φυσιολογικά, το FreeBSD χρησιμοποιεί το μέγεθος της εγκατεστημένης μνήμης για να καθορίσει μια σειρά από παραμέτρους του πυρήνα, όπως το μέγιστο αριθμό αρχείων που μπορεί να είναι ταυτόχρονα ανοιχτά. Σε συστήματα με περισσότερη από 1GB μνήμη, αυτός ο μηχανισμός "αυτόματης ρύθμισης μεγεθών" ίσως επιλέξει τιμές οι οποίες να είναι πολύ υψηλές. Κατά την εκκίνηση, ο πυρήνας εκχωρεί διάφορους πίνακες και άλλες δομές, οι οποίες καταλαμβάνουν τον περισσότερο διαθέσιμο χώρο του. Αργότερα, καθώς το σύστημα λειτουργεί, ο πυρήνας δεν έχει άλλο χώρο για δυναμικές εκχωρήσεις μνήμης, και δημιουργείται panic.

Δημιουργήστε το δικό σας πυρήνα, και προσθέστε την επιλογή `VM_KMEM_SIZE_MAX` στο αρχείο ρυθμίσεων του, ώστε να αυξήσετε το μέγιστο μέγεθος σε 400 MB (`options VM_KMEM_SIZE_MAX=419430400`). Τα 400 MB φαίνεται να επαρκούν για μηχανήματα με μέγεθος μνήμης ως 6 GB.

5.12. Το σύστημα μου δεν έχει 1GB RAM, και πάλι όμως το FreeBSD δημιουργεί panic με το μήνυμα kmem_map too small!

Το panic δείχνει ότι το σύστημα έχει μείνει από εικονική μνήμη για προσωρινή αποθήκευση δεδομένων δικτύου (network buffers, και ειδικότερα mbuf clusters). Μπορείτε να αυξήσετε το μέγεθος της εικονικής μνήμης που διατίθεται για mbuf clusters, ακολουθώντας τις οδηγίες στην

ενότητα Όρια Δικτύου του Εγχειριδίου.

5.13. Γιατί παίρνω το μήνυμα λάθους /kernel: proc: table is full;

Ο πυρήνας του FreeBSD επιτρέπει κάθε χρονική στιγμή την ύπαρξη ενός συγκεκριμένου αριθμού διεργασιών. Ο αριθμός αυτός βασίζεται στην επιλογή `MAXUSERS` του πυρήνα. Το `MAXUSERS` επηρεάζει επίσης και άλλα όρια μέσα στον πυρήνα, όπως η προσωρινή μνήμη του δικτύου (network buffers) (δείτε την προηγούμενη ερώτηση). Αν το μηχάνημα σας λειτουργεί σε υψηλό φορτίο, ίσως έχει νόημα να αυξήσετε την επιλογή `MAXUSERS`. Με τον τρόπο αυτό, μαζί με το μέγιστο αριθμό διεργασιών, θα αυξηθούν και άλλα όρια του συστήματος.

Για να ρυθμίσετε την τιμή του `MAXUSERS`, δείτε την ενότητα Όρια Αρχείων/Διεργασιών του Εγχειριδίου. (Αν και η ενότητα αυτή αναφέρεται σε ανοιχτά αρχεία, τα ίδια όρια ισχύουν και για τις διεργασίες.)

Αν το μηχάνημα σας λειτουργεί σε χαμηλό φορτίο, αλλά εκτελεί μεγάλο αριθμό διεργασιών, μπορείτε απλώς να ρυθμίσετε τον αριθμό τους αλλάζοντας την τιμή της μεταβλητής `kern.maxproc`. Αν πρέπει να ρυθμίσετε αυτή τη μεταβλητή, θα πρέπει να την ορίσετε στο αρχείο /boot/loader.conf. Η ρύθμιση δεν θα ισχύσει μέχρι να επανεκκινήσετε το σύστημα. Για περισσότερες πληροφορίες σχετικά με τις μεταβλητές του πυρήνα, δείτε τις σελίδες manual loader.conf(5) και sysctl.conf(5). Αν όλες αυτές οι διεργασίες εκτελούνται από ένα μόνο χρήστη, θα πρέπει επίσης να ρυθμίσετε την τιμή της μεταβλητής `kern.maxprocperuid` ώστε να είναι κατά ένα μικρότερη από την νέα τιμή της `kern.maxproc`. (Πρέπει να είναι κατά ένα μικρότερη, γιατί υπάρχει πάντα ένα πρόγραμμα συστήματος, το init(8), που πρέπει να εκτελείται συνέχεια.).

Για να γίνει μόνιμη μια αλλαγή ενός sysctl, τοποθετήστε την κατάλληλη τιμή στο αρχείο /etc/sysctl.conf. Περισσότερες πληροφορίες για τη ρύθμιση του συστήματος με την χρήση του sysctl(8), μπορείτε να βρείτε στην ενότητα Ρυθμίσεις μέσω sysctl του Εγχειριδίου.

5.14. Γιατί παίρνω το μήνυμα λάθους CMAP busy όταν επανεκκινώ με νέο πυρήνα;

Η λογική του συστήματος που προσπαθεί να ανιχνεύσει τυχόν παλιές εκδόσεις των αρχείων /var/db/kvm_*.db κάποιες φορές αποτυγχάνει, και η χρήση ανόμοιων εκδόσεων μπορεί σε ορισμένες περιπτώσεις να οδηγήσει σε panic.

Αν σας συμβεί αυτό, επανεκκινήστε σε κατάσταση ενός χρήστη (single user) και γράψτε:

```
# rm /var/db/kvm_*.db
```

5.15. Τι σημαίνει το μήνυμα ahc0: brkadrint, Illegal Host Access at seqaddr 0x0;

Υπάρχει μια διένεξη με την κάρτα Ultrastor SCSI Host Adapter.

Κατά τη διάρκεια της διαδικασίας εκκίνησης, εισέλθετε στο μενού ρυθμίσεων του πυρήνα και απενεργοποιήστε τη συσκευή uha0, η οποία είναι αυτή που προκαλεί το πρόβλημα.

5.16. Όταν ξεκινώ το σύστημα μου παίρνω το λάθος ahc0: illegal cable configuration. Η καλωδίωση μου είναι σωστή. Τι συμβαίνει;

Η μητρική πλακέτα σας δεν έχει τα απαιτούμενα εξωτερικά κυκλώματα ώστε να υποστηρίζει αυτόματο τερματισμό του διαύλου SCSI. Αντί να βασίζεστε στον αυτόματο τερματισμό, δηλώστε στο SCSI BIOS τον σωστό τερματισμό για τη διάταξη συσκευών που έχετε. Το πρόγραμμα οδήγησης του AIC7XXX δεν μπορεί να καθορίσει αν είναι διαθέσιμο το κύκλωμα που χρησιμοποιείται για την ανίχνευση του καλωδίου (άρα και του αυτόματου τερματισμού). Το πρόγραμμα οδήγησης υποθέτει ότι υπάρχει υποστήριξη, εφόσον οι ρυθμίσεις που περιέχονται στη σειριακή EEPROM αναφέρουν "αυτόματο τερματισμό". Συχνά, χωρίς το εξωτερικό κύκλωμα ανίχνευσης του καλωδίου, το πρόγραμμα οδήγησης θα ρυθμίζει λανθασμένα τον τερματισμό, κάτι που μπορεί να δημιουργήσει πρόβλημα στην αξιοπιστία του διαύλου SCSI.

5.17. Γιατί το Sendmail δίνει το μήνυμα λάθους mail loops back to myself;

Αυτό περιγράφεται στο sendmail FAQ όπως φαίνεται παρακάτω:

```
        * Παίρνω μηνύματα λάθους "Local configuration error" όπως το:

        553 relay.domain.net config error: mail loops back to myself
        554 user@domain.net... Local configuration error

        Πως μπορώ να επιλύσω το πρόβλημα;

        Έχετε ζητήσει να κατευθύνετε το mail προς το domain (π.χ. domain.net)
        προς κάποιο συγκεκριμένο μηχάνημα (στην περίπτωση αυτή, το
        relay.domain.net) χρησιμοποιώντας μια εγγραφή MX, αλλά το μηχάνημα
        που κάνει την ανακατεύθυνση δεν αναγνωρίζει τον εαυτό του ως
        domain.net.  Προσθέστε το domain.net στο /etc/mail/local-host-names
        (αν χρησιμοποιείτε το FEATURE(use_cw_file)) ή προσθέστε
        "Cw domain.net" στο /etc/mail/sendmail.cf.
```

Η τρέχουσα έκδοση του sendmail FAQ δεν συντηρείται πλέον με κάθε έκδοση του sendmail. Ωστόσο, δημοσιεύεται ανά τακτά διαστήματα στις λίστες comp.mail.sendmail, comp.mail.misc, comp.mail.smail, comp.answers, και news.answers. Μπορείτε επίσης να λάβετε αντίγραφο μέσω email, στέλνοντας ένα μήνυμα στο mail-server@rtfm.mit.edu με την εντολή `send usenet/news.answers/mail/sendmail-faq` στο κύριο μέρος του μηνύματος.

5.18. Γιατί δεν συμπεριφέρονται σωστά οι εφαρμογές πλήρους οθόνης σε απομακρυσμένα μηχανήματα;

Είναι πιθανόν το απομακρυσμένο μηχάνημα να ρυθμίζει τον τύπο του τερματικού σας σε κάτι διαφορετικό από τον τύπο `cons25` που απαιτείται από την κονσόλα του FreeBSD.

Υπάρχουν διάφοροι τρόποι για να παρακάμψετε αυτό το πρόβλημα:

- Μετά την είσοδο σας στο απομακρυσμένο μηχάνημα, ορίστε την μεταβλητή TERM του κελύφους σε `ansi` ή `sco`, εφόσον το απομακρυσμένο μηχάνημα μπορεί να λειτουργήσει με αυτά τα είδη τερματικών.
- Στην κονσόλα του FreeBSD, χρησιμοποιήστε κάποιο εξομοιωτή τερματικού VT100, όπως το screen. Το screen σας δίνει τη δυνατότητα να έχετε πολλαπλές συνεδρίες από ένα μόνο τερματικό, και είναι έτσι και αλλιώς χρήσιμο πρόγραμμα. Κάθε παράθυρο του screen συμπεριφέρεται ως τερματικό του VT100, έτσι η μεταβλητή TERM στον απομακρυσμένο υπολογιστή θα πρέπει να ρυθμιστεί σε `vt100`.
- Εγκαταστήστε την καταχώρηση `cons25` στη βάση δεδομένων τερματικών του απομακρυσμένου υπολογιστή. Ο τρόπος για να γίνει αυτό, εξαρτάται από το λειτουργικό σύστημα του απομακρυσμένου υπολογιστή. Φυσιολογικά, θα βρείτε αυτές τις πληροφορίες στα εγχειρίδια διαχείρισης συστήματος του απομακρυσμένου μηχανήματος.
- Στο τοπικό σας FreeBSD μηχάνημα, χρησιμοποιήστε τον X server και κάντε login στο απομακρυσμένο μηχάνημα χρησιμοποιώντας κάποιο εξομοιωτή τερματικού όπως το `xterm` ή το `rxvt`. Στην περίπτωση αυτή, θα πρέπει στο απομακρυσμένο μηχάνημα να ρυθμίσετε την μεταβλητή TERM σε `xterm` ή `vt100`.

5.19. Γιατί το μηχάνημα μου δείχνει το μήνυμα calcru: negative time...;

Αυτό μπορεί να συμβεί από διάφορες αιτίες που σχετίζονται με interrupts, τόσο στο υλικό όσο και στο λογισμικό. Μπορεί να οφείλεται σε προβλήματα (bugs) αλλά μπορεί επίσης να προκληθεί εξαιτίας της φύσης κάποιων συσκευών. Ένας συνηθισμένος τρόπος πρόκλησης του προβλήματος, είναι η εκτέλεση εφαρμογών TCP/IP με μεγάλο MTU μέσω της παράλληλης θύρας. Μπορεί επίσης να προκληθεί από κάποιους επιταχυντές γραφικών, και στην περίπτωση αυτή το πρώτο πράγμα που θα πρέπει να ελέγξετε είναι η ρύθμιση interrupt της αντίστοιχης κάρτας.

Παρενέργεια αυτού του προβλήματος είναι ο απότομος τερματισμός διεργασιών με το μήνυμα "SIGXCPU exceeded cpu time limit".

Αν το πρόβλημα δεν μπορεί να λυθεί με διαφορετικό τρόπο, η λύση είναι να ορίσετε την παρακάτω μεταβλητή του sysctl:

```
# sysctl -w kern.timecounter.method=1
```

Η επιλογή `-w` του sysctl(8) θεωρείται παρωχημένη και αγνοείται σιωπηλά από το FreeBSD 4.4-RELEASE και μετά. Μπορείτε με ασφάλεια να το παραλείψετε κατά τη

ρύθμιση των επιλογών με την `sysctl` όπως φαίνεται παραπάνω.

Το παραπάνω θα έχει επίδραση στην απόδοση, αλλά σε σχέση με την αιτία του προβλήματος, μάλλον δεν θα το παρατηρήσετε. Αν το πρόβλημα επιμένει, διατηρήστε την τιμή του sysctl στο ένα, και ρυθμίστε την επιλογή `NTIMECOUNTER` στον πυρήνα σας, σε ολοένα αυξανόμενες τιμές. Αν φτάσετε την τιμή `NTIMECOUNTER=20` και το πρόβλημα δεν έχει λυθεί, τα interrupts στο μηχάνημα σας είναι πολύ προβληματικά και ακατάλληλα για ακριβή ρύθμιση της ώρας.

5.20. Γιατί η PnP κάρτα μου δεν ανιχνεύεται πλέον (ή ανιχνεύεται ως unknown) μετά την αναβάθμιση σε FreeBSD 4.X;

Το FreeBSD 4.X ακολουθεί πλέον αρκετά πιο πιστά το πρότυπο *PnP* και αυτό δημιουργεί ορισμένες φορές την παρενέργεια να μη λειτουργούν κάποιες συσκευές PnP (π.χ. κάρτες ήχου και εσωτερικά modems) οι οποίες ωστόσο λειτουργούσαν στο FreeBSD 3.X.

Οι λόγοι για την συμπεριφορά αυτή, εξηγούνται στο ακόλουθο e-mail, το οποίο στάλθηκε στη λίστα freebsd-questions από τον Peter Wemm, ως απάντηση σε ερώτηση σχετικά με ένα εσωτερικό modem το οποίο δεν ήταν ανιχνεύσιμο από το σύστημα μετά από αναβάθμιση σε FreeBSD 4.X (τα σχόλια μέσα σε `[]` έχουν προστεθεί για να γίνει πιο κατανοητό το αντικείμενο της συζήτησης).

Το περιεχόμενο αυτής της παράθεσης έχει ανανεωθεί σε σχέση με το αρχικό κείμενο.

> Το PNP bios το προ-ρύθμισε [το modem] και το άφησε στην περιοχή διευθύνσεων των θυρών, και έτσι [στην έκδοση 3.X] η παλαιού τύπου ανίχνευση ISA το "βρήκε" εκεί.
>
> Στην έκδοση 4.0, ο κώδικας διαχείρισης του ISA, είναι πολύ περισσότερο προσανατολισμένος στο PnP μοντέλο. Στο 3.X ήταν δυνατόν η ανίχνευση ISA να εντοπίσει μια "χαμένη" συσκευή και έπειτα η PNP συσκευή να ταιριάξει και να αποτύχει η ρύθμιση της λόγω διένεξης πόρων. Έτσι, απενεργοποιούνται αρχικά οι προγραμματιζόμενες κάρτες, ώστε να μη συμβεί αυτή η διπλή ανίχνευση. Αυτό επίσης σημαίνει ότι η ανίχνευση πρέπει να γνωρίζει τα PnP ids των υποστηριζόμενων συσκευών. Είναι στις προθέσεις μας να κάνουμε τη διαδικασία αυτή περισσότερη προσβάσιμη στους χρήστες.

Για να λειτουργήσει ξανά η συσκευή, πρέπει να βρεθεί το PNP id της και να προστεθεί στη λίστα των ανιχνεύσεων ISA που χρησιμοποιούνται για την αναγνώριση PnP συσκευών. Αυτό μπορεί να γίνει με τη χρήση της pnpinfo(8) για την ανίχνευση της συσκευής, για παράδειγμα αυτή είναι η έξοδος της pnpinfo(8) για ένα εσωτερικό modem:

```
# pnpinfo
Checking for Plug-n-Play devices...

Card assigned CSN #1
Vendor ID PMC2430 (0x3024a341), Serial Number 0xffffffff
PnP Version 1.0, Vendor Version 0
Device Description: Pace 56 Voice Internal Plug & Play Modem

Logical Device ID: PMC2430 0x3024a341 #0
    Device supports I/O Range Check
TAG Start DF
    I/O Range 0x3f8 .. 0x3f8, alignment 0x8, len 0x8
    [16-bit addr]
    IRQ: 4  - only one type (true/edge)
```

```
TAG End DF
End Tag

Successfully got 31 resources, 1 logical fdevs
-- card select # 0x0001

CSN PMC2430 (0x3024a341), Serial Number 0xffffffff

Logical device #0
IO:  0x03e8 0x03e8 0x03e8 0x03e8 0x03e8 0x03e8 0x03e8 0x03e8
IRQ 5 0
DMA 4 0
IO range check 0x00 activate 0x01
```

Οι πληροφορίες που απαιτούνται, βρίσκονται στη γραμμή "Vendor ID", στην αρχή της εξόδου. Ο δεκαεξαδικός αριθμός στις παρενθέσεις (στο παράδειγμα μας 0x3024a341) είναι το PnP id ενώ το αλφαριθμητικό που βρίσκεται ακριβώς πριν από αυτόν είναι ένα μοναδικό ASCII αναγνωριστικό.

Εναλλακτικά, αν το pnpinfo(8) δεν δείχνει την ζητούμενη κάρτα, μπορείτε να χρησιμοποιήσετε το pciconf(8). Παρακάτω φαίνεται ένα μέρος της εξόδου της `pciconf -vl` για ένα κύκλωμα ήχου ενσωματωμένου στη μητρική:

```
# pciconf -vl
chip1@pci0:31:5:        class=0x040100 card=0x00931028 chip=0x24158086 rev=0x02 hdr
=0x00
    vendor   = 'Intel Corporation'
    device   = '82801AA 8xx Chipset AC'97 Audio Controller'
    class    = multimedia
    subclass = audio
```

Εδώ, θα χρησιμοποιούσαμε την τιμή του `chip`, "0x24158086".

Η πληροφορία αυτή (Vendor ID ή τιμή chip) θα πρέπει να προστεθεί στο αρχείο /usr/src/sys/isa/sio.c.

Θα πρέπει πρώτα να κρατήσετε ένα αντίγραφο ασφαλείας του sio.c, για την περίπτωση που κάτι πάει στραβά. Επίσης, θα χρειαστείτε το αντίγραφο για να δημιουργήσετε ένα patch το οποίο θα καταθέσετε με την αναφορά προβλήματος (PR) που θα μας στείλετε (και θα μας στείλετε PR, έτσι;). Κατόπιν επεξεργαστείτε το sio.c και ψάξτε για τη γραμμή

```
static struct isa_pnp_id sio_ids[] = {
```

έπειτα μετακινηθείτε προς τα κάτω για να βρείτε το σωστό μέρος να προσθέσετε την καταχώρηση της συσκευής σας. Οι καταχωρήσεις φαίνονται όπως παρακάτω και είναι ταξινομημένες κατά το αλφαριθμητικό ASCII Vendor ID το οποίο θα πρέπει να περιληφθεί στο σχόλιο στο δεξιό μέρος της γραμμής μαζί με όλη την περιγραφή *Device Description* (αν χωράει, αλλιώς μέρος της) από την έξοδο της pnpinfo(8):

```
{0x0f804f3f, NULL},     /* OZO800f - Zoom 2812 (56k Modem) */
{0x39804f3f, NULL},     /* OZO8039 - Zoom 56k flex */
{0x3024a341, NULL},     /* PMC2430 - Pace 56 Voice Internal Modem */
{0x1000eb49, NULL},     /* ROK0010 - Rockwell ? */
{0x5002734a, NULL},     /* RSS0250 - 5614Jx3(G) Internal Modem */
```

Προσθέστε το δεκαεξαδικό Vendor ID για τη συσκευή σας στο σωστό μέρος, αποθηκεύστε το αρχείο, αναδημιουργήστε τον πυρήνα σας, και επανεκκινήστε. Θα πρέπει τώρα η συσκευή σας να βρεθεί ως συσκευή `sio` όπως συνέβαινε και με το FreeBSD 3.X

5.21. Γιατί παίρνω το λάθος nlist failed όταν εκτελώ, για παράδειγμα, το top ή το systat;

Το πρόβλημα είναι ότι η εφαρμογή που προσπαθείτε να εκτελέσετε ψάχνει για ένα συγκεκριμένο σύμβολο στον πυρήνα, αλλά για κάποιο λόγο δεν μπορεί να το εντοπίσει. Το σφάλμα αυτό μπορεί να οφείλεται σε δύο προβλήματα:

- Ο πυρήνας σας και τα υπόλοιπα βασικά προγράμματα (userland) δεν είναι σε συγχρονισμό (π.χ. έχετε δημιουργήσει νέο πυρήνα, αλλά δεν εκτελέσατε `installworld`, ή αντίστροφα), με αποτέλεσμα ο πίνακας συμβόλων να είναι διαφορετικός από αυτόν που πιστεύει η εφαρμογή. Αν πρόκειται για αυτή την περίπτωση, απλώς ολοκληρώστε τη διαδικασία αναβάθμισης (δείτε το /usr/src/UPDATING για τη σωστή ακολουθία εντολών).
- Δεν χρησιμοποιείτε το `/boot/loader` για να φορτώσετε τον πυρήνα σας, αλλά τον φορτώνετε απευθείας από το boot2 (δείτε το boot(8)). Αν και δεν είναι λάθος να παρακάμψετε τον `/boot/loader`, σε γενικές γραμμές το πρόγραμμα αυτό τα καταφέρνει καλύτερα στο να διαθέτει τα σύμβολα του πυρήνα στις εφαρμογές χρήστη.

5.22. Γιατί παίρνει τόσο χρόνο να συνδεθώ με τον υπολογιστή μου μέσω ssh ή telnet;

Το σύμπτωμα: Υπάρχει μεγάλη καθυστέρηση μεταξύ της στιγμής που αποκαθίσταται η TCP σύνδεση και της στιγμής που το πρόγραμμα στη μεριά του πελάτη ζητάει τον κωδικό πρόσβασης (ή στην περίπτωση του telnet(1), της στιγμής που εμφανίζεται η προτροπή login).

Το πρόβλημα: Το πιο πιθανό είναι ότι η καθυστέρηση οφείλεται στην προσπάθεια που καταβάλλει το λογισμικό στη μεριά του εξυπηρετητή να βρει το όνομα του μηχανήματος - πελάτη από την IP διεύθυνση του. Οι περισσότεροι εξυπηρετητές, συμπεριλαμβανομένων του Telnet και SSH που έρχονται με το FreeBSD, λειτουργούν με αυτό τον τρόπο, ώστε μεταξύ άλλων, να αποθηκεύσουν το όνομα του μηχανήματος σε ένα αρχείο καταγραφής για μελλοντική αναφορά από τον διαχειριστή.

Η θεραπεία: Αν το πρόβλημα προκύπτει κάθε φορά που συνδέεστε από τον υπολογιστή σας (τον πελάτη) σε οποιοδήποτε εξυπηρετητή, το πρόβλημα βρίσκεται στον πελάτη. Με τον ίδιο τρόπο, αν το πρόβλημα συμβαίνει μόνο όταν κάποιος συνδέεται στον υπολογιστή σας (τον εξυπηρετητή), το πρόβλημα βρίσκεται στον εξυπηρετητή.

Αν το πρόβλημα είναι στον πελάτη, η μόνη θεραπεία είναι να διορθώσετε το DNS, ώστε ο εξυπηρετητής να μπορεί να το βρει. Αν το πρόβλημα εμφανίζεται στο τοπικό σας δίκτυο, θεωρείστε το πρόβλημα στον εξυπηρετητή και συνεχίστε την ανάγνωση. Αντίθετα, αν το πρόβλημα εμφανίζεται σε συνδέσεις μέσω Internet, κατά πάσα πιθανότητα θα χρειαστεί να επικοινωνήσετε με τον ISP σας και να ζητήσετε να σας το διορθώσει.

Αν το πρόβλημα είναι με τον εξυπηρετητή, και εμφανίζεται στο τοπικό σας δίκτυο, θα πρέπει να τον ρυθμίσετε ώστε να μπορεί να εκτελεί αναζητήσεις τύπου διεύθυνση σε όνομα, για την τοπική περιοχή διευθύνσεων σας. Δείτε τις σελίδες manual των hosts(5) και named(8) για περισσότερες πληροφορίες. Αν το πρόβλημα εμφανίζεται στις συνδέσεις μέσω Internet, μπορεί να οφείλεται σε κακή λειτουργία του resolver στον εξυπηρετητή σας. Για να το ελέγξετε, δοκιμάστε να βρείτε κάποιο άλλο μηχάνημα, για παράδειγμα το `www.yahoo.com`. Αν ούτε αυτό δουλεύει, εκεί βρίσκεται το πρόβλημα σας.

Μετά από μια νέα εγκατάσταση του FreeBSD είναι επίσης πιθανό να λείπουν οι πληροφορίες για τον τομέα (domain) και τον εξυπηρετητή ονομάτων (nameserver) από το αρχείο /etc/resolv.conf. Αυτό επίσης θα προκαλέσει καθυστέρηση στο SSH, καθώς η επιλογή "UseDNS" έχει ως προεπιλεγμένη την τιμή "yes" στο αρχείο ρυθμίσεων sshd_config στον κατάλογο /etc/ssh. Αν είναι αυτή η αιτία του προβλήματος, θα πρέπει είτε να συμπληρώσετε τις απαιτούμενες πληροφορίες στο /etc/resolv.conf ή να θέσετε το "UseDNS" στο "no" στο αρχείο sshd_config ως προσωρινή λύση.

5.23. Ποια είναι η έννοια του stray (περιπλανώμενου) IRQ;

Τα stray IRQs είναι σημάδια προβλημάτων υλικού που χρησιμοποιεί IRQs, ειδικότερα σχετίζεται με υλικό που κατά τη μέση του κύκλου αναγνώρισης (acknowledge cycle) του interrupt, σταματάει να μεταδίδει την αντίστοιχη αίτηση διακοπής.

Έχετε τρεις επιλογές για να αντιμετωπίσετε αυτό το πρόβλημα:

- Ανεχθείτε τις προειδοποιήσεις. Έτσι και αλλιώς, μετά τις 5 πρώτες, δεν θα δείτε άλλες.
- Σταματήστε εντελώς τις προειδοποιήσεις, αλλάζοντας το 5 σε 0 στην `isa_strayintr()`.
- Σταματήστε τις προειδοποιήσεις εγκαθιστώντας υλικό για την παράλληλη πόρτα που να χρησιμοποιεί το IRQ 7 και το αντίστοιχο για αυτό πρόγραμμα οδήγησης PPP (αυτό συμβαίνει στα περισσότερα συστήματα) και εγκαταστήστε ένα οδηγό IDE ή άλλο υλικό που να χρησιμοποιεί το irq 15 μαζί με το κατάλληλο πρόγραμμα οδήγησης του.

5.24. Γιατί βλέπω συνέχεια το μήνυμα file: table is full στο dmesg;

Το μήνυμα αυτό σημαίνει ότι έχετε εξαντλήσει τον αριθμό των διαθέσιμων περιγραφέων αρχείων (file descriptors) στο σύστημα σας. Παρακαλούμε δείτε το kern.maxfiles τμήμα στο κεφάλαιο Ρύθμιση Ορίων Πυρήνα του Εγχειριδίου, για ερμηνεία και επίλυση του προβλήματος.

5.25. Γιατί το ρολόι στο φορητό μου υπολογιστή δεν κρατάει την σωστή ώρα;

Ο φορητός υπολογιστής σας έχει δύο ή περισσότερα ρολόγια, και το FreeBSD έχει επιλέξει να χρησιμοποιήσει το λάθος.

Εκτελέστε την dmesg(8), και ελέγξτε για γραμμές που περιέχουν την λέξη `Timecounter`. Η τελευταία από τις γραμμές που θα εκτυπωθεί δείχνει το ρολόι που επιλέχθηκε από το FreeBSD και σχεδόν σίγουρα θα είναι το `TSC`.

```
# dmesg | grep Timecounter
Timecounter "i8254"  frequency 1193182 Hz
Timecounter "TSC"  frequency 595573479 Hz
```

Μπορείτε να το επιβεβαιώσετε αυτό, ελέγχοντας την τιμή του `kern.timecounter.hardware` sysctl(3).

```
# sysctl kern.timecounter.hardware
kern.timecounter.hardware: TSC
```

Το BIOS ίσως να τροποποιεί την τιμή του ρολογιού TSC- ενδεχομένως για να αλλάξει την ταχύτητα του επεξεργαστή όταν λειτουργεί με μπαταρίες, ή όταν εισέρχεται σε κατάσταση χαμηλής κατανάλωσης, αλλά το FreeBSD δεν γνωρίζει για αυτές τις αλλαγές και φαίνεται να κερδίζει ή να χάνει χρόνο.

Στο παράδειγμα μας, είναι επίσης διαθέσιμο το ρολόι `i8254` και μπορείτε να το επιλέξετε γράφοντας το όνομα του στο sysctl(3) `kern.timecounter.hardware`.

```
# sysctl -w kern.timecounter.hardware=i8254
kern.timecounter.hardware: TSC -> i8254
```

Ο φορητός υπολογιστής σας θα πρέπει τώρα να είναι πιο ακριβής στην τήρηση του χρόνου.

Για να παραμείνει η αλλαγή αυτή σε κάθε εκκίνηση, προσθέστε την παρακάτω γραμμή στο /etc/sysctl.conf.

```
kern.timecounter.hardware=i8254
```

5.26. Γιατί ο φορητός μου υπολογιστής δεν αναγνωρίζει σωστά τις κάρτες τύπου PC card;

Το πρόβλημα είναι κοινό σε φορητά που εκκινούν περισσότερα από ένα λειτουργικά συστήματα. Ορισμένα μη-BSD λειτουργικά συστήματα αφήνουν τις PC cards σε μη-προβλέψιμη κατάσταση. Η εντολή `pccardd` σε αυτή την περίπτωση, ανιχνεύει την κάρτα ως `"(null)""(null)"` αντί για το πραγματικό της μοντέλο.

Πρέπει να αποσυνδέσετε εντελώς την τροφοδοσία από την θύρα PC card ώστε το υλικό να επανέλθει στην αρχική του κατάσταση. Απενεργοποιήστε πλήρως τον φορητό υπολογιστή σας. (Μην τον βάλετε σε κατάσταση αναμονής ή ύπνου, θα πρέπει να απενεργοποιηθεί εντελώς.) Περιμένετε για λίγα λεπτά και επανεκκινήστε. Θα πρέπει τώρα η PC card να λειτουργεί κανονικά.

Το υλικό κάποιων φορητών υπολογιστών στην πραγματικότητα παραμένει ενεργό, ακόμα και όταν υποτίθεται ότι ο υπολογιστής είναι ανενεργός. Αν το παραπάνω δεν έχει το επιθυμητό αποτέλεσμα, τερματίστε τη λειτουργία του υπολογιστή σας, αφαιρέστε την μπαταρία, περιμένετε λίγο, τοποθετήστε ξανά την μπαταρία και επανεκκινήστε.

5.27. Γιατί ο φορτωτής εκκίνησης του FreeBSD δείχνει το μήνυμα λάθους Read error και σταματάει μετά την οθόνη του BIOS;

Ο φορτωτής εκκίνησης του FreeBSD δεν αναγνωρίζει σωστά την γεωμετρία του σκληρού δίσκου. Μπορείτε να την ρυθμίσετε χειροκίνητα μέσα από την fdisk κατά την δημιουργία ή τροποποίηση του slice του FreeBSD.

Μπορείτε να βρείτε τις σωστές τιμές για την γεωμετρία του οδηγού στο BIOS του μηχανήματος. Ψάξτε για τον αριθμό των κυλίνδρων, κεφαλών και τομέων για τον οδηγό που θέλετε.

Μέσα από την fdisk του sysinstall(8), πιέστε το `G` για να ορίσετε την γεωμετρία του οδηγού.

Θα εμφανιστεί ένας διάλογος που θα ζητάει τον αριθμό των κυλίνδρων, κεφαλών και τομέων. Πληκτρολογήστε τους αριθμούς που βρήκατε από το BIOS, χωρίζοντας τους με κανονικές καθέτους. Για παράδειγμα, για 5000 κυλίνδρους, 250 κεφαλές και 60 τομείς, θα γράφαμε `5000/250/60`.

Πιέστε enter για να ορίσετε τις τιμές, και έπειτα το `W` για να γράψετε το νέο πίνακα κατατμήσεων στον οδηγό.

5.28. Ένα άλλο λειτουργικό σύστημα κατέστρεψε τον διαχειριστή εκκίνησης μου. Πως μπορώ να τον αποκαταστήσω;

Θα πρέπει να εισέλθετε στο sysinstall(8) και να επιλέξετε Configure και κατόπιν Fdisk. Επιλέξτε το δίσκο στον οποίο βρίσκεται κανονικά ο Φορτωτής Εκκίνησης χρησιμοποιώντας το πλήκτρο `space`. Πιέστε το `W` για να γράψετε τις αλλαγές στον οδηγό. Θα εμφανιστεί μια προτροπή που θα σας ρωτάει ποιο φορτωτή εκκίνησης να εγκαταστήσει. Κάντε την αντίστοιχη επιλογή και ο φορτωτής εκκίνησης θα αποκατασταθεί.

5.29. Τι σημαίνει το μήνυμα λάθους swap_pager: indefinite wait buffer:;

Σημαίνει ότι μια διαδικασία προσπαθεί να γράψει μια σελίδα μνήμης στο δίσκο, και η απόπειρα αυτή έχει κολλήσει προσπαθώντας να αποκτήσει πρόσβαση στο δίσκο για περισσότερο από 20 δευτερόλεπτα. Αυτό μπορεί να συμβεί από χαλασμένους τομείς στο σκληρό δίσκο, προβληματικά καλώδια, ή άλλο υλικό το οποίο να σχετίζεται με I/O. Αν πρόκειται για προβληματικό δίσκο, θα δείτε επίσης και αντίστοιχα μηνύματα στο /var/log/messages και στην έξοδο της εντολής `dmesg`. Διαφορετικά, ελέγξτε τις συνδέσεις και τα καλώδια σας.

5.30. Τι είναι τα σφάλματα UDMA ICRC, και πως μπορώ να τα διορθώσω;

Το πρόγραμμα οδήγησης ata(4) αναφέρει σφάλματα τύπου "UDMA ICRC" όταν εντοπίσει πρόβλημα στην ορθότητα των δεδομένων σε μια μεταφορά DMA από ή προς τον οδηγό. Το πρόγραμμα οδήγησης θα προσπαθήσει να επαναλάβει τη μεταφορά μερικές φορές. Αν όλες οι απόπειρες αποτύχουν, θα αλλάξει την κατάσταση επικοινωνίας της συσκευής από DMA σε PIO, η οποία είναι πιο αργή.

Το πρόβλημα μπορεί να προκληθεί από πολλούς παράγοντες, αν και ο πιο συνηθισμένος είναι η προβληματική ή λανθασμένη καλωδίωση. Ελέγξτε ότι τα καλώδια ATA δεν έχουν υποστεί ζημιά, και ότι είναι κατάλληλων προδιαγραφών για την κατάσταση λειτουργίας Ultra DMA που χρησιμοποιείτε. Αν χρησιμοποιείτε αφαιρούμενα συρτάρια δίσκων, θα πρέπει επίσης να είναι συμβατά. Βεβαιωθείτε ότι υπάρχει καλή επαφή σε όλες τις συνδέσεις. Έχουν επίσης αναφερθεί προβλήματα όταν γίνεται εγκατάσταση ενός παλιού οδηγού στο ίδιο κανάλι DMA με ένα δίσκο Ultra DMA 66 (ή πιο γρήγορο). Τέλος, τα λάθη αυτά μπορεί να σημαίνουν ότι ο δίσκος πρόκειται σύντομα να χαλάσει. Οι περισσότεροι κατασκευαστές δίσκων παρέχουν λογισμικό ελέγχου για τους οδηγούς τους, ελέγξτε λοιπόν το δίσκο σας, και αν χρειάζεται, πάρτε αντίγραφο των δεδομένων σας και αντικαταστήστε τον.

Μπορείτε να χρησιμοποιήσετε το βοηθητικό πρόγραμμα atacontrol(8) για να δείτε και να επιλέξετε την κατάσταση λειτουργίας DMA ή PIO που χρησιμοποιείται από κάθε συσκευή ATA. Πιο συγκεκριμένα, η εντολή `atacontrol mode channel` θα σας δείξει την κατάσταση λειτουργίας των συσκευών ενός συγκεκριμένου καναλιού ATA, όπου το πρωτεύον κανάλι έχει την αρίθμηση 0 κ.ο.κ.

5.31. Τι είναι το lock order reversal;

Ο Robert Watson <rwatson@FreeBSD.org> απάντησε με σαφήνεια αυτή την ερώτηση στην λίστα ηλεκτρονική λίστα της έκδοσης FreeBSD-CURRENT, σε μια συζήτηση με τίτλο "lock order reversals - τι σημαίνουν;"

Από τον Robert Watson <rwatson@FreeBSD.org> στη λίστα ηλεκτρονική λίστα της έκδοσης FreeBSD-CURRENT, στις 14 Δεκεμβρίου 2003 Οι προειδοποιήσεις αυτές προέρχονται από το Witness, ένα διαγνωστικό σύστημα για κλειδώματα κατά τη λειτουργία (run-time lock) το οποίο βρίσκεται στους πυρήνες -CURRENT του FreeBSD (αλλά αφαιρείται στις επίσημες εκδόσεις). Μπορείτε να διαβάσετε περισσότερα για το Witness και τις δυνατότητες του, στη σελίδα manual witness(4). Μεταξύ άλλων το Witness επαληθεύει τη σειρά των run-time locks χρησιμοποιώντας ένα συνδυασμό από ενσωματωμένες σειρές κλειδωμάτων καθώς και από τη σειρά που ανιχνεύεται κατά την εκτέλεση, και παράγει προειδοποιήσεις στην κονσόλα όταν παραβιάζονται. Σκοπός αυτής της λειτουργίας είναι να ανιχνεύονται πιθανά deadlocks τα οποία μπορεί να οφείλονται σε παραβιάσεις της σειράς των κλειδωμάτων. Είναι αξιοσημείωτο ότι το Witness είναι κάπως συντηρητικό, και είναι πιθανόν να δώσει λάθος προειδοποιήσεις. Στην περίπτωση που το Witness αναφέρει ένα πραγματικό πρόβλημα με την σειρά των κλειδωμάτων, είναι σαν να λέει "αν ήσασταν άτυχος, θα σας είχε συμβεί deadlock σε αυτό το σημείο". Υπάρχουν κάποιες γνωστές περιπτώσεις "λανθασμένης διάγνωσης" για τις οποίες χρειάζεται να δημιουργήσουμε καλύτερη τεκμηρίωση ώστε να αποφύγουμε και τις περιττές αναφορές σφαλμάτων. Οι λιγότερο γνωστές περιπτώσεις οφείλονται περισσότερο σε νέα κλειδώματα, καθώς οι αντιστροφές στη σειρά των κλειδωμάτων διορθώνονται γρήγορα επειδή το Witness είναι πάντα απασχολημένο και δημιουργεί συνέχεια νέες προειδοποιήσεις :-).

Αυτό που αποκαλούμε "λανθασμένη διάγνωση" δημιουργείται στην πραγματικότητα όταν το Witness βρίσκει κάποιο πολύ πιο σοβαρό λάθος. Τέτοια λάθη είναι τυπικά το σφάλμα σελίδας (page fault) ή λανθασμένα δεδομένα στη μνήμη μέσα στον πυρήνα, ή τέλος σύγκρουση ονομασίας με κάποια mutexes.

Δείτε την σελίδα του Bjoern Zeeb σχετικά με τις αντιστροφές κλειδωμάτων για την κατάσταση των γνωστών αντιστροφών.

5.32. Τι σημαίνει το μήνυμα Called ... with the following non-sleepable locks held;

Σημαίνει ότι κλήθηκε μια συνάρτηση με δυνατότητα sleep ενώ την ίδια στιγμή ήταν ενεργό κάποιο κλείδωμα mutex (ή αντίστοιχο χωρίς δυνατότητα sleep).

Ο λόγος για τον οποίο αυτό είναι λάθος είναι επειδή τα mutexes δεν προορίζονται να κρατούνται για μεγάλα χρονικά διαστήματα. Είναι μόνο για τη συντήρηση μικρών περιόδων συγχρονισμού. Αυτή η προγραμματιστική συμφωνία επιτρέπει στους οδηγούς συσκευών να χρησιμοποιούν mutexes για να συγχρονίζονται με τα υπόλοιπα προγράμματα του πυρήνα κατά την διάρκεια των interrupts. Τα interrupts (στο FreeBSD) δεν μπορούν να περιέλθουν σε κατάσταση sleep. Για το λόγο αυτό είναι απαραίτητο να μην μπλοκάρεται ο πυρήνας για μεγάλο διάστημα από κάποιο υποσύστημα που κρατάει ένα mutex.

Για να εντοπιστούν αυτά τα λάθη, μπορούν να προστεθούν υποθέσεις (assertions) στον πυρήνα οι οποίες αλληλεπιδρούν με το υποσύστημα witness για να δώσουν ένα προειδοποιητικό μήνυμα (ή μήνυμα λάθους, ανάλογα με τις ρυθμίσεις του συστήματος) όταν γίνεται μια κλήση η οποία πιθανώς να δημιουργεί μπλοκάρισμα την στιγμή που κρατιέται ένα mutex.

Εν συντομία, αυτού του είδους οι προειδοποιήσεις δεν είναι συνήθως μοιραίες, αλλά υπό ορισμένες ατυχείς προϋποθέσεις, μπορεί να προκαλέσουν ανεπιθύμητα φαινόμενα τα οποία κυμαίνονται από μια στιγμιαία πτώση στην απόκριση του συστήματος, μέχρι πλήρης κατάρρευση.

5.33. Γιατί η διαδικασία buildworld/installworld σταματάει με το μήνυμα touch: not found;

Το μήνυμα αυτό δεν σημαίνει ότι σας λείπει το βοηθητικό πρόγραμμα touch(1). Το λάθος αυτό προκαλείται συνήθως από λανθασμένη, μελλοντική, σήμανση ημερομηνίας των αρχείων. Αν το ρολόι CMOS του υπολογιστή σας είναι ρυθμισμένο για τοπική ώρα, πρέπει να εκτελέσετε την εντολή `adjkerntz -i` για να ρυθμίσετε το ρολόι του πυρήνα όταν εκκινείτε σε κατάσταση λειτουργίας ενός χρήστη.

Chapter 6. Εμπορικές Εφαρμογές

This section is still very sparse, though we are hoping, of course, that companies will add to it! :) The FreeBSD group has no financial interest in any of the companies listed here but simply lists them as a public service (and feels that commercial interest in FreeBSD can have very positive effects on FreeBSD's long-term viability). We encourage commercial software vendors to send their entries here for inclusion. See the Vendors page for a longer list.

6.1. Where can I get an Office Suite for FreeBSD?

The open-source OpenOffice.org office suite works natively on FreeBSD. The Linux® version of StarOffice, the value-added closed-source version of OpenOffice.org, also works on FreeBSD.

FreeBSD also includes a variety of text editors, spreadsheets, and drawing programs in the Ports Collection.

6.2. Where can I get Motif® for FreeBSD?

The Open Group has released the source code to Motif® 2.2.2. You can install the `open-motif` package, or compile it from ports. Refer to the ports section of the Handbook for more information on how to do this.

The Open Motif® distribution only allows redistribution if it is running on an open source operating system.

In addition, there are commercial distributions of the Motif® software available. These, however, are not for free, but their license allows them to be used in closed-source software. Contact Apps2go for the least expensive ELF Motif® 2.1.20 distribution for FreeBSD (either i386™ or Alpha).

There are two distributions, the "development edition" and the "runtime edition" (for much less). These distributions includes:

- OSF/Motif® manager, xmbind, panner, wsm.
- Development kit with uil, mrm, xm, xmcxx, include and Imake files.
- Static and dynamic ELF libraries.
- Demonstration applets.

Be sure to specify that you want the FreeBSD version of Motif® when ordering (do not forget to mention the architecture you want too)! Versions for NetBSD and OpenBSD are also sold by *Apps2go*. This is currently a FTP only download.

More info

Apps2go WWW page

or

sales@apps2go.com or support@apps2go.com

or

phone (817) 431 8775 or +1 817 431-8775

Contact Xi Graphics for an a.out Motif® 2.0 distribution for FreeBSD.

This distribution includes:

- OSF/Motif® manager, xmbind, panner, wsm.
- Development kit with uil, mrm, xm, xmcxx, include and Imake files.
- Static and dynamic libraries (for use with FreeBSD 2.2.8 and earlier).
- Demonstration applets.
- Preformatted manual pages.

Be sure to specify that you want the FreeBSD version of Motif® when ordering! Versions for BSDI and Linux® are also sold by *Xi Graphics*. This is currently a 4 diskette set... in the future this will change to a unified CD distribution like their CDE.

6.3. Where can I get CDE for FreeBSD?

Xi Graphics used to sell CDE for FreeBSD, but no longer do.

KDE is an open source X11 desktop which is similar to CDE in many respects. You might also like the look and feel of xfce. KDE and xfce are both in the ports system.

6.4. Are there any commercial high-performance X servers?

Yes, Xi Graphics sells Accelerated-X products for FreeBSD and other Intel based systems.

The Xi Graphics offering is a high performance X Server that offers easy configuration, support for multiple concurrent video boards and is distributed in binary form only, in a unified diskette distribution for FreeBSD and Linux®. Xi Graphics also offers a high performance X Server tailored for laptop support.

There is a free "compatibility demo" of version 5.0 available.

Xi Graphics also sells Motif® and CDE for FreeBSD (see above).

More info

Xi Graphics WWW page

or

sales@xig.com or support@xig.com

or

phone (800) 946 7433 or +1 303 298-7478.

6.5. Are there any Database systems for FreeBSD?

Yes! See the Commercial Vendors section of FreeBSD's Web site.

Also see the Databases section of the Ports collection.

6.6. Can I run Oracle® on FreeBSD?

Yes. The following pages tell you exactly how to set up Linux®-Oracle® on FreeBSD:

- http://www.unixcities.com/oracle/index.html
- http://www.shadowcom.net/freebsd-oracle9i/

Chapter 7. Εφαρμογές Τελικού Χρήστη

7.1. So, where are all the user applications?

Please take a look at the ports page for info on software packages ported to FreeBSD. The list currently tops 36000 and is growing daily, so come back to check often or subscribe to the `freebsd-announce`mailing list for periodic updates on new entries.

Most ports should work on the 4.X, 5.X, and 6.X branches. Each time a FreeBSD release is made, a snapshot of the ports tree at the time of release in also included in the ports/ directory.

We also support the concept of a "package", essentially no more than a compressed binary distribution with a little extra intelligence embedded in it for doing whatever custom installation work is required. A package can be installed and uninstalled again easily without having to know the gory details of which files it includes.

Use the package installation menu in /stand/sysinstall (under the post-configuration menu item) or invoke the pkg_add(1) command on the specific package files you are interested in installing. Package files can usually be identified by their .tgz or .tbz suffix and CDROM distribution people will have a packages/All directory on their CD which contains such files. They can also be downloaded over the net for various versions of FreeBSD at the following locations:

for 4.X-RELEASE/4-STABLE

ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/i386/packages-4-stable/

for 5.X-RELEASE/5-STABLE

ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/i386/packages-5-stable

for 6.X-RELEASE/6-STABLE

ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/i386/packages-6-stable

for 7-CURRENT

ftp://ftp.FreeBSD.org/pub/FreeBSD/ports/i386/packages-7-current

or your nearest local mirror site.

Note that all ports may not be available as packages since new ones are constantly being added. It is always a good idea to check back periodically to see which packages are available at the ftp.FreeBSD.org master site.

7.2. How do I configure INN (Internet News) for my machine?

After installing the news/inn package or port, an excellent place to start is Dave Barr's INN Page where you will find the INN FAQ.

7.3. Does FreeBSD support Java™?

Yes. Please see http://www.FreeBSD.org/java/.

7.4. Why can I not build this port on my 4.X-STABLE machine?

If you are running a FreeBSD version that lags significantly behind -CURRENT or -STABLE, you may need to update your ports collection; see the Keeping Up section of the Porter's Handbook for further information on how to do this. If you are up to date, then someone might have committed a change to the port which works for -CURRENT but which broke the port for -STABLE. Please submit a bug report on this with the send-pr(1) command, since the ports collection is supposed to work for both the -CURRENT and -STABLE branches.

7.5. I just tried to build INDEX using make index, and it failed. Why?

First, always make sure that you have a completely up-to-date Ports Collection. Errors that affect building INDEX from an up-to-date copy of the Ports Collection are high-visibility and are thus almost always fixed immediately.

However, if you are up-to-date, perhaps you are seeing another problem. `make index` has a known bug in dealing with incomplete copies of the Ports Collection. It assumes that you have a local copy of every single port that every other port that you have a local copy of depends on. To explain, if you have a copy of foo/bar on your disk, and foo/bar depends on baz/quux, then you must also have a copy of baz/quux on your disk, and the ports baz/quux depends on, and so on. Otherwise, `make index` has insufficient information to create its dependency tree.

This is particularly a problem for FreeBSD users who utilize cvsup(1) to track the Ports Collection but choose not to install certain categories by specifying them in refuse. In theory, one should be able to refuse categories, but in practice there are too many ports that depend on ports in other categories. Until someone comes up with a solution for this problem, the general rule is that if you want to build INDEX, you must have a complete copy of the Ports Collection.

There are rare cases where INDEX will not build due to odd cases involving `WITH* or WITHOUT*` variables being set in make.conf. If you suspect that this is the case, please try to make INDEX with those Makevars turned off before reporting it to ηλεκτρονική λίστα των FreeBSD ports.

7.6. Why is CVSup not integrated in the main FreeBSD tree?

The FreeBSD base system is designed as self-hosting - it should be possible to build the whole operating system starting with a very limited set of tools. Thus, the actual build tools needed to compile the FreeBSD sources are bundled with the sources themselves. This includes a C compiler (gcc(1)), make(1), awk(1), and similar tools.

Since CVSup is written in Modula-3, adding it to the FreeBSD base system would also require adding and maintaining a Modula-3 compiler. This would lead to both an increase in the disk space consumed by the FreeBSD sources and additional maintenance work. Thus, it is much easier for both the developers and users to keep CVSup as a separate port, which can be easily installed as a package bundled on the FreeBSD installation CDs.

7.7. I updated the sources, now how do I update my installed ports?

FreeBSD does not include a port upgrading tool, but it does have some tools to make the upgrade process somewhat easier. You can also install additional tools to simplify port handling.

The pkg_version(1) command can generate a script that will update installed ports to the latest version in the ports tree.

```
# pkg_version -c > /tmp/myscript
```

The output script *must* be edited by hand before you use it. Recent versions of pkg_version(1) force this by inserting an exit(1) at the beginning of the script.

You should save the output of the script, as it will note packages that depend on the one that has been updated. These may or may not need to be updated as well. The usual case where they need to be updated is that a shared library has changed version numbers, so the ports that used that library need to be rebuilt to use the new version.

> Beginning with FreeBSD 5.0 (and higher revisions), pkg_version(1) no longer supports the `-c` option.

If you have the disk space, you can use the `portupgrade` tool to automate all of this. `portupgrade` includes various tools to simplify package handling. It is available under ports-mgmt/portupgrade. Since it is written in Ruby, `portupgrade` is an unlikely candidate for integration with the main FreeBSD tree. That should not stop anyone from using it, however.

If your system is up full time, the periodic(8) system can be used to generate a weekly list of ports that might need updating by setting `weekly_status_pkg_enable="YES"` in /etc/periodic.conf.

7.8. Why is /bin/sh so minimal? Why does FreeBSD not use bash or another shell?

Because POSIX® says that there shall be such a shell.

The more complicated answer: many people need to write shell scripts which will be portable across many systems. That is why POSIX® specifies the shell and utility commands in great detail. Most scripts are written in Bourne shell, and because several important programming interfaces (make(1), system(3), popen(3), and analogues in higher-level scripting languages like Perl and Tcl) are specified to use the Bourne shell to interpret commands. Because the Bourne shell is so often

and widely used, it is important for it to be quick to start, be deterministic in its behavior, and have a small memory footprint.

The existing implementation is our best effort at meeting as many of these requirements simultaneously as we can. In order to keep `/bin/sh` small, we have not provided many of the convenience features that other shells have. That is why the Ports Collection includes more featureful shells like bash, scsh, tcsh, and zsh. (You can compare for yourself the memory utilization of all these shells by looking at the "VSZ" and "RSS" columns in a `ps -u` listing.)

7.9. Why do getenv(3) and Opera take so long to start?

The usual answer is that DNS on your system is misconfigured. Both getenv(3) and Opera perform DNS checks when starting up. The browser will not appear on your desktop until the program either gets a response or determines that the system has no network connection.

7.10. I updated parts of the Ports Collection using CVSup, and now many ports fail to build with mysterious error messages! What happened? Is the Ports Collection broken in some major way?

If you only update parts of the Ports Collection, using one of its CVSup subcollections and not the `ports-all` CVSup collection, you should *always* update the `ports-base` subcollection too! The reasons are described in the Handbook.

7.11. How do I create audio CDs from my MIDI files?

To create audio CDs from MIDI files, first install audio/timidity from ports then install manually the GUS patches set by Eric A. Welsh, available at http://www.stardate.bc.ca/eawpatches/html/default.htm. After timidity++ has been installed properly, midi files may be converted to wav files with the following command line:

```
% timidity -Ow -s 44100 -o /tmp/juke/01.wav 01.mid
```

The wav files can then be converted to other formats or burned onto audio CDs, as described in the FreeBSD Handbook.

Chapter 8. Ρύθμιση Πυρήνα

8.1. I would like to customize my kernel. Is it difficult?

Not at all! Check out the kernel config section of the Handbook.

> We recommend that you make a dated snapshot of your new /kernel called /kernel.YYMMDD after you get it working properly. Also back up your new /modules directory to /modules.YYMMDD. That way, if you make a mistake the next time you play with your configuration you can boot the backup kernel instead of having to fall back to kernel.GENERIC. This is particularly important if you are now booting from a controller that GENERIC does not support.

8.2. My kernel compiles fail because _hw_float is missing. How do I solve this problem?

You probably removed npx0 (see npx(4)) from your kernel configuration file because you do not have a math co-processor. The npx0 device is *MANDATORY*. Somewhere inside your hardware lies a device that provides hardware floating-point support, even if it is no longer a separate device as used in the good old 386 days. You *must* include the npx0 device. Even if you manage to build a kernel without npx0 support, it will not boot anyway.

8.3. Why is my kernel so big (over 10MB)?

Chances are, you compiled your kernel in *debug mode*. Kernels built in debug mode contain many symbols that are used for debugging, thus greatly increasing the size of the kernel. Note that there will be little or no performance decrease from running a debug kernel, and it is useful to keep one around in case of a system panic.

However, if you are running low on disk space, or you simply do not want to run a debug kernel, make sure that both of the following are true:

- You do not have a line in your kernel configuration file that reads:

  ```
  makeoptions DEBUG=-g
  ```

- You are not running config(8) with the `-g` option.

Either of the above settings will cause your kernel to be built in debug mode. As long as you make sure you follow the steps above, you can build your kernel normally, and you should notice a fairly large size decrease; most kernels tend to be around 1.5MB to 2MB.

8.4. Why do I get interrupt conflicts with multi-port serial code?

When I compile a kernel with multi-port serial code, it tells me that only the first port is probed and the rest skipped due to interrupt conflicts. How do I fix this?

The problem here is that FreeBSD has code built-in to keep the kernel from getting trashed due to hardware or software conflicts. The way to fix this is to leave out the IRQ settings on all but one port. Here is an example:

```
#
# Multiport high-speed serial line - 16550 UARTS
#
device sio2 at isa? port 0x2a0 tty irq 5 flags 0x501 vector siointr
device sio3 at isa? port 0x2a8 tty flags 0x501 vector siointr
device sio4 at isa? port 0x2b0 tty flags 0x501 vector siointr
device sio5 at isa? port 0x2b8 tty flags 0x501 vector siointr
```

8.5. Why does every kernel I try to build fail to compile, even GENERIC?

There are a number of possible causes for this problem. They are, in no particular order:

- You are not using the new `make buildkernel` and `make installkernel` targets, and your source tree is different from the one used to build the currently running system (e.g., you are compiling 4.3-RELEASE on a 4.0-RELEASE system). If you are attempting an upgrade, please read the /usr/src/UPDATING file, paying particular attention to the "COMMON ITEMS" section at the end.
- You are using the new `make buildkernel` and `make installkernel` targets, but you failed to assert the completion of the `make buildworld` target. The `make buildkernel` target relies on files generated by the `make buildworld` target to complete its job correctly.
- Even if you are trying to build FreeBSD-STABLE, it is possible that you fetched the source tree at a time when it was either being modified, or broken for other reasons; only releases are absolutely guaranteed to be buildable, although FreeBSD-STABLE builds fine the majority of the time. If you have not already done so, try re-fetching the source tree and see if the problem goes away. Try using a different server in case the one you are using is having problems.

8.6. How can I verify which scheduler is in use on a running system?

If you are running FreeBSD version 5.2.1 or earlier, check for the existence of the `kern.quantum` sysctl. If you have it, you should see something like this:

```
% sysctl kern.quantum
```

```
kern.sched.quantum: 99960
```

If the `kern.quantum` sysctl exists, you are using the 4BSD scheduler. If not, you will get an error printed by sysctl(8) (which you can safely ignore):

```
% sysctl kern.sched.quantum
sysctl: unknown oid 'kern.sched.quantum'
```

In FreeBSD version 5.3-RELEASE and later, the name of the scheduler currently being used is directly available as the value of the `kern.sched.name` sysctl:

```
% sysctl kern.sched.name
kern.sched.name: 4BSD
```

8.7. What is kern.quantum?

`kern.quantum` is the maximum number of ticks a process can run without being preempted. It is specific to the 4BSD scheduler, so you can use its presence or absence to determine which scheduler is in use. In FreeBSD 5.X or later `kern.quantum` has been renamed to `kern.sched.quantum`.

8.8. What is kern.sched.quantum?

See What is kern.quantum?

Chapter 9. Δίσκοι, Συστήματα Αρχείων και Φορτωτές Εκκίνησης

9.1. How do I move my system over to my huge new disk?

The best way is to reinstall the OS on the new disk, then move the user data over. This is highly recommended if you have been tracking -STABLE for more than one release, or have updated a release instead of installing a new one. You can install booteasy on both disks with boot0cfg(8), and dual boot them until you are happy with the new configuration. Skip the next paragraph to find out how to move the data after doing this.

Should you decide not to do a fresh install, you need to partition and label the new disk with either /stand/sysinstall, or fdisk(8) and disklabel(8). You should also install booteasy on both disks with boot0cfg(8), so that you can dual boot to the old or new system after the copying is done.

Now you have the new disk set up, and are ready to move the data. Unfortunately, you cannot just blindly copy the data. Things like device files (in /dev), flags, and links tend to screw that up. You need to use tools that understand these things, which means dump(8). Although it is suggested that you move the data in single user mode, it is not required.

You should never use anything but dump(8) and restore(8) to move the root filesystem. The tar(1) command may work - then again, it may not. You should also use dump(8) and restore(8) if you are moving a single partition to another empty partition. The sequence of steps to use dump to move a partitions data to a new partition is:

1. newfs the new partition.
2. mount it on a temporary mount point.
3. cd to that directory.
4. dump the old partition, piping output to the new one.

For example, if you are going to move root to /dev/ad1s1a, with /mnt as the temporary mount point, it is:

```
# newfs /dev/ad1s1a
# mount /dev/ad1s1a /mnt
# cd /mnt
# dump 0af - / | restore xf -
```

Rearranging your partitions with dump takes a bit more work. To merge a partition like /var into its parent, create the new partition large enough for both, move the parent partition as described above, then move the child partition into the empty directory that the first move created:

```
# newfs /dev/ad1s1a
# mount /dev/ad1s1a /mnt
# cd /mnt
# dump 0af - / | restore xf -
# cd var
# dump 0af - /var | restore xf -
```

To split a directory from its parent, say putting /var on its own partition when it was not before, create both partitions, then mount the child partition on the appropriate directory in the temporary mount point, then move the old single partition:

```
# newfs /dev/ad1s1a
# newfs /dev/ad1s1d
# mount /dev/ad1s1a /mnt
# mkdir /mnt/var
# mount /dev/ad1s1d /mnt/var
# cd /mnt
# dump 0af - / | restore xf -
```

You might prefer cpio(1), pax(1), tar(1) to dump(8) for user data. At the time of this writing, these are known to lose file flag information, so use them with caution.

9.2. Will a dangerously dedicated disk endanger my health?

The installation procedure allows you to chose two different methods in partitioning your hard disk(s). The default way makes it compatible with other operating systems on the same machine, by using fdisk table entries (called "slices" in FreeBSD), with a FreeBSD slice that employs partitions of its own. Optionally, one can chose to install a boot-selector to switch between the possible operating systems on the disk(s). The alternative uses the entire disk for FreeBSD, and makes no attempt to be compatible with other operating systems.

So why it is called "dangerous"? A disk in this mode does not contain what normal PC utilities would consider a valid fdisk table. Depending on how well they have been designed, they might complain at you once they are getting in contact with such a disk, or even worse, they might damage the BSD bootstrap without even asking or notifying you. In addition, the "dangerously dedicated" disk's layout is known to confuse many BIOSes, including those from AWARD (e.g. as found in HP Netserver and Micronics systems as well as many others) and Symbios/NCR (for the popular 53C8xx range of SCSI controllers). This is not a complete list, there are more. Symptoms of this confusion include the `read error` message printed by the FreeBSD bootstrap when it cannot find itself, as well as system lockups when booting.

Why have this mode at all then? It only saves a few kbytes of disk space, and it can cause real problems for a new installation. "Dangerously dedicated" mode's origins lie in a desire to avoid one of the most common problems plaguing new FreeBSD installers - matching the BIOS "geometry" numbers for a disk to the disk itself.

"Geometry" is an outdated concept, but one still at the heart of the PC's BIOS and its interaction with disks. When the FreeBSD installer creates slices, it has to record the location of these slices on the disk in a fashion that corresponds with the way the BIOS expects to find them. If it gets it wrong, you will not be able to boot.

"Dangerously dedicated" mode tries to work around this by making the problem simpler. In some cases, it gets it right. But it is meant to be used as a last-ditch alternative - there are better ways to solve the problem 99 times out of 100.

So, how do you avoid the need for "DD" mode when you are installing? Start by making a note of the geometry that your BIOS claims to be using for your disks. You can arrange to have the kernel print this as it boots by specifying `-v` at the `boot:` prompt, or using `boot -v` in the loader. Just before the installer starts, the kernel will print a list of BIOS geometries. Do not panic - wait for the installer to start and then use scrollback to read the numbers. Typically the BIOS disk units will be in the same order that FreeBSD lists your disks, first IDE, then SCSI.

When you are slicing up your disk, check that the disk geometry displayed in the FDISK screen is correct (ie. it matches the BIOS numbers); if it is wrong, use the `g` key to fix it. You may have to do this if there is absolutely nothing on the disk, or if the disk has been moved from another system. Note that this is only an issue with the disk that you are going to boot from; FreeBSD will sort itself out just fine with any other disks you may have.

Once you have got the BIOS and FreeBSD agreeing about the geometry of the disk, your problems are almost guaranteed to be over, and with no need for "DD" mode at all. If, however, you are still greeted with the dreaded `read error` message when you try to boot, it is time to cross your fingers and go for it - there is nothing left to lose.

To return a "dangerously dedicated" disk for normal PC use, there are basically two options. The first is, you write enough NULL bytes over the MBR to make any subsequent installation believe this to be a blank disk. You can do this for example with

```
# dd if=/dev/zero of=/dev/rda0 count=15
```

Alternatively, the undocumented DOS "feature"

```
C:\> fdisk /mbr
```

will to install a new master boot record as well, thus clobbering the BSD bootstrap.

9.3. Which partitions can safely use Soft Updates? I have heard that Soft Updates on / can cause problems.

Short answer: you can usually use Soft Updates safely on all partitions.

Long answer: There used to be some concern over using Soft Updates on the root partition. Soft Updates has two characteristics that caused this. First, a Soft Updates partition has a small chance

of losing data during a system crash. (The partition will not be corrupted; the data will simply be lost.) Also, Soft Updates can cause temporary space shortages.

When using Soft Updates, the kernel can take up to thirty seconds to actually write changes to the physical disk. If you delete a large file, the file still resides on disk until the kernel actually performs the deletion. This can cause a very simple race condition. Suppose you delete one large file and immediately create another large file. The first large file is not yet actually removed from the physical disk, so the disk might not have enough room for the second large file. You get an error that the partition does not have enough space, although you know perfectly well that you just released a large chunk of space! When you try again mere seconds later, the file creation works as you expect. This has left more than one user scratching his head and doubting his sanity, the FreeBSD filesystem, or both.

If a system should crash after the kernel accepts a chunk of data for writing to disk, but before that data is actually written out, data could be lost or corrupted. This risk is extremely small, but generally manageable. Use of IDE write caching greatly increases this risk; it is strongly recommended that you disable IDE write caching when using Soft Updates.

These issues affect all partitions using Soft Updates. So, what does this mean for the root partition?

Vital information on the root partition changes very rarely. Files such as /kernel and the contents of /etc only change during system maintenance, or when users change their passwords. If the system crashed during the thirty-second window after such a change is made, it is possible that data could be lost. This risk is negligible for most applications, but you should be aware that it exists. If your system cannot tolerate this much risk, do not use Soft Updates on the root filesystem!

/ is traditionally one of the smallest partitions. By default, FreeBSD puts the /tmp directory on /. If you have a busy /tmp, you might see intermittent space problems. Symlinking /tmp to /var/tmp will solve this problem.

9.4. What is inappropriate about my ccd?

The symptom of this is:

```
# ccdconfig -C
ccdconfig: ioctl (CCDIOCSET): /dev/ccd0c: Inappropriate file type or format
```

This usually happens when you are trying to concatenate the `c` partitions, which default to type `unused`. The ccd driver requires the underlying partition type to be FS_BSDFFS. Edit the disklabel of the disks you are trying to concatenate and change the types of partitions to `4.2BSD`.

9.5. Why can I not edit the disklabel on my ccd?

The symptom of this is:

```
# disklabel ccd0
(it prints something sensible here, so let us try to edit it)
```

```
# disklabel -e ccd0
(edit, save, quit)
disklabel: ioctl DIOCWDINFO: No disk label on disk;
use "disklabel -r" to install initial label
```

This is because the disklabel returned by ccd is actually a "fake" one that is not really on the disk. You can solve this problem by writing it back explicitly, as in:

```
# disklabel ccd0 > /tmp/disklabel.tmp
# disklabel -Rr ccd0 /tmp/disklabel.tmp
# disklabel -e ccd0
(this will work now)
```

9.6. Can I mount other foreign filesystems under FreeBSD?

FreeBSD supports a variety of other filesystems.

Digital UNIX

UFS CDROMs can be mounted directly on FreeBSD. Mounting disk partitions from Digital UNIX and other systems that support UFS may be more complex, depending on the details of the disk partitioning for the operating system in question.

Linux®

FreeBSD supports `ext2fs` partitions. See mount_ext2fs(8) for more information.

Windows NT®

FreeBSD includes a read-only NTFS driver. For more information, see mount_ntfs(8).

FAT

FreeBSD includes a read-write FAT driver. For more information, see mount_msdosfs(8).

ReiserFS

FreeBSD includes a read-only ReiserFS driver. For more information, see mount_reiserfs(8).

FreeBSD also supports network filesystems such as NFS (see mount_nfs(8)), NetWare (see mount_nwfs(8)), and Microsoft-style SMB filesystems (see mount_smbfs(8)).

9.7. How do I mount a secondary DOS partition?

The secondary DOS partitions are found after ALL the primary partitions. For example, if you have an "E" partition as the second DOS partition on the second SCSI drive, you need to create the special files for "slice 5" in /dev, then mount /dev/da1s5:

```
# cd /dev
# sh MAKEDEV da1s5
```

```
# mount -t msdosfs /dev/da1s5 /dos/e
```

You can omit this step if you are running FreeBSD 5.0-RELEASE or newer with devfs(5) enabled.

9.8. Is there a cryptographic filesystem for FreeBSD?

Yes. FreeBSD 5.0 includes gbde(8), and FreeBSD 6.0 added geli(8). For earlier releases, see the security/cfs port.

9.9. How can I use the Windows NT® loader to boot FreeBSD?

The general idea is that you copy the first sector of your native root FreeBSD partition into a file in the DOS/Windows NT® partition. Assuming you name that file something like c:\bootsect.bsd (inspired by c:\bootsect.dos), you can then edit the c:\boot.ini file to come up with something like this:

```
[boot loader]
timeout=30
default=multi(0)disk(0)rdisk(0)partition(1)\WINDOWS
[operating systems]
multi(0)disk(0)rdisk(0)partition(1)\WINDOWS="Windows NT"
C:\BOOTSECT.BSD="FreeBSD"
C:\="DOS"
```

If FreeBSD is installed on the same disk as the Windows NT® boot partition simply copy /boot/boot1 to C:\BOOTSECT.BSD. However, if FreeBSD is installed on a different disk /boot/boot1 will not work, /boot/boot0 is needed.

/boot/boot0 needs to be installed using sysinstall by selecting the FreeBSD boot manager on the screen which asks if you wish to use a boot manager. This is because /boot/boot0 has the partition table area filled with NULL characters but sysinstall copies the partition table before copying /boot/boot0 to the MBR.

Do not simply copy /boot/boot0 instead of /boot/boot1; you will overwrite your partition table and render your computer un-bootable!

When the FreeBSD boot manager runs it records the last OS booted by setting the active flag on the partition table entry for that OS and then writes the whole 512-bytes of itself back to the MBR so if you just copy /boot/boot0 to C:\BOOTSECT.BSD then it writes an empty partition table, with the active flag set on one entry, to the MBR.

9.10. How do I boot FreeBSD and Linux® from LILO?

If you have FreeBSD and Linux® on the same disk, just follow LILO's installation instructions for booting a non-Linux® operating system. Very briefly, these are:

Boot Linux®, and add the following lines to /etc/lilo.conf:

```
other=/dev/hda2
        table=/dev/hda
        label=FreeBSD
```

(the above assumes that your FreeBSD slice is known to Linux® as /dev/hda2; tailor to suit your setup). Then, run `lilo` as `root` and you should be done.

If FreeBSD resides on another disk, you need to add `loader=/boot/chain.b` to the LILO entry. For example:

```
other=/dev/dab4
        table=/dev/dab
        loader=/boot/chain.b
        label=FreeBSD
```

In some cases you may need to specify the BIOS drive number to the FreeBSD boot loader to successfully boot off the second disk. For example, if your FreeBSD SCSI disk is probed by BIOS as BIOS disk 1, at the FreeBSD boot loader prompt you need to specify:

```
Boot: 1:da(0,a)/kernel
```

You can configure boot(8) to automatically do this for you at boot time.

The Linux®+FreeBSD mini-HOWTO is a good reference for FreeBSD and Linux® interoperability issues.

9.11. How do I boot FreeBSD and Linux® using GRUB

Booting FreeBSD using GRUB is very simple. Just add the following to your configuration file /boot/grub/grub.conf.

```
title FreeBSD 6.1
    root (hd0,a)
    kernel /boot/loader
```

Where `hd0,a` points to your root partition on the first disk. If you need to specify which slice number should be used, use something like this `(hd0,2,a)`. By default, if the slice number is omitted, GRUB searches the first slice which has `'a'` partition.

9.12. How do I boot FreeBSD and Linux® using BootEasy?

Install LILO at the start of your Linux® boot partition instead of in the Master Boot Record. You can then boot LILO from BootEasy.

If you are running Windows® 95 and Linux® this is recommended anyway, to make it simpler to get Linux® booting again if you should need to reinstall Windows® 95 (which is a Jealous Operating System, and will bear no other Operating Systems in the Master Boot Record).

9.13. How do I change the boot prompt from ??? to something more meaningful?

You can not do that with the standard boot manager without rewriting it. There are a number of other boot managers in the sysutils ports category that provide this functionality.

9.14. I have a new removable drive, how do I use it?

Whether it is a removable drive like a Zip® or an EZ drive (or even a floppy, if you want to use it that way), or a new hard disk, once it is installed and recognized by the system, and you have your cartridge/floppy/whatever slotted in, things are pretty much the same for all devices.

(this section is based on Mark Mayo's ZIP FAQ)

If it is a ZIP drive or a floppy, you have already got a DOS filesystem on it, you can use a command like this:

```
# mount -t msdosfs /dev/fd0c /floppy
```

if it is a floppy, or this:

```
# mount -t msdosfs /dev/da2s4 /zip
```

for a ZIP disk with the factory configuration.

For other disks, see how they are laid out using fdisk(8) or sysinstall(8).

The rest of the examples will be for a ZIP drive on da2, the third SCSI disk.

Unless it is a floppy, or a removable you plan on sharing with other people, it is probably a better idea to stick a BSD filesystem on it. You will get long filename support, at least a 2X improvement in performance, and a lot more stability. First, you need to redo the DOS-level partitions/filesystems. You can either use fdisk(8) or /stand/sysinstall, or for a small drive that you do not want to bother with multiple operating system support on, just blow away the whole FAT partition table (slices) and just use the BSD partitioning:

```
# dd if=/dev/zero of=/dev/rda2 count=2
# disklabel -Brw da2 auto
```

You can use disklabel or /stand/sysinstall to create multiple BSD partitions. You will certainly want to do this if you are adding swap space on a fixed disk, but it is probably irrelevant on a removable drive like a ZIP.

Finally, create a new filesystem, this one is on our ZIP drive using the whole disk:

```
# newfs /dev/rda2c
```

and mount it:

```
# mount /dev/da2c /zip
```

and it is probably a good idea to add a line like this to /etc/fstab (see fstab(5)) so you can just type `mount /zip` in the future:

```
/dev/da2c /zip ffs rw,noauto 0 0
```

9.15. Why do I get Incorrect super block when mounting a CDROM?

You have to tell mount(8) the type of the device that you want to mount. This is described in the Handbook section on optical media, specifically the section Using Data CDs.

9.16. Why do I get Device not configured when mounting a CDROM?

This generally means that there is no CDROM in the CDROM drive, or the drive is not visible on the bus. Please see the Using Data CDs section of the Handbook for a detailed discussion of this issue.

9.17. Why do all non-English characters in filenames show up as ? on my CDs when mounted in FreeBSD?

Your CDROM probably uses the "Joliet" extension for storing information about files and directories. This is discussed in the Handbook chapter on creating and using CDROMs, specifically the section on Using Data CDROMs.

9.18. I burned a CD under FreeBSD and now I can not read it under any other operating system. Why?

You most likely burned a raw file to your CD, rather than creating an ISO 9660 filesystem. Take a look at the Handbook chapter on creating CDROMs, particularly the section on burning raw data CDs.

9.19. How can I create an image of a data CD?

This is discussed in the Handbook section on duplicating data CDs. For more on working with CDROMs, see the Creating CDs Section in the Storage chapter in the Handbook.

9.20. Why can I not mount an audio CD?

If you try to mount an audio CD, you will get an error like `cd9660: /dev/acd0c: Invalid argument`. This is because `mount` only works on filesystems. Audio CDs do not have filesystems; they just have data. You need a program that reads audio CDs, such as the audio/xmcd port.

9.21. How do I mount a multi-session CD?

By default, mount(8) will attempt to mount the last data track (session) of a CD. If you would like to load an earlier session, you must use the `-s` command line argument. Please see mount_cd9660(8) for specific examples.

9.22. How do I let ordinary users mount floppies, CDROMs and other removable media?

Ordinary users can be permitted to mount devices. Here is how:

1. As `root` set the sysctl variable `vfs.usermount` to `1`.

   ```
   # sysctl -w vfs.usermount=1
   ```

2. As `root` assign the appropriate permissions to the block device associated with the removable media.

   For example, to allow users to mount the first floppy drive, use:

   ```
   # chmod 666 /dev/fd0
   ```

   To allow users in the group `operator` to mount the CDROM drive, use:

   ```
   # chgrp operator /dev/acd0c
   ```

```
# chmod 640 /dev/acd0c
```

3. If you are running FreeBSD 5.X or later, you will need to alter /etc/devfs.conf to make these changes permanent across reboots.

   As `root`, add the necessary lines to /etc/devfs.conf. For example, to allow users to mount the first floppy drive add:

   ```
   # Allow all users to mount the floppy disk.
   own       /dev/fd0    root:operator
   perm      /dev/fd0    0666
   ```

   To allow users in the group `operator` to mount the CD-ROM drive add:

   ```
   # Allow members of the group operator to mount CD-ROMs.
   own       /dev/acd0   root:operator
   perm      /dev/acd0   0660
   ```

4. Finally, add the line `vfs.usermount=1` to the file /etc/sysctl.conf so that it is reset at system boot time.

All users can now mount the floppy /dev/fd0 onto a directory that they own:

```
% mkdir ~/my-mount-point
% mount -t msdosfs /dev/fd0 ~/my-mount-point
```

Users in group `operator` can now mount the CDROM /dev/acd0c onto a directory that they own:

```
% mkdir ~/my-mount-point
% mount -t cd9660 /dev/acd0c ~/my-mount-point
```

Unmounting the device is simple:

```
% umount ~/my-mount-point
```

Enabling `vfs.usermount`, however, has negative security implications. A better way to access MS-DOS® formatted media is to use the emulators/mtools package in the ports collection.

> The device name used in the previous examples must be changed according to your configuration.

9.23. The du and df commands show different amounts of disk space available. What is going on?

You need to understand what `du` and `df` really do. `du` goes through the directory tree, measures how large each file is, and presents the totals. `df` just asks the filesystem how much space it has left. They seem to be the same thing, but a file without a directory entry will affect `df` but not `du`.

When a program is using a file, and you delete the file, the file is not really removed from the filesystem until the program stops using it. The file is immediately deleted from the directory listing, however. You can see this easily enough with a program such as `more`. Assume you have a file large enough that its presence affects the output of `du` and `df`. (Since disks can be so large today, this might be a *very* large file!) If you delete this file while using `more` on it, `more` does not immediately choke and complain that it cannot view the file. The entry is simply removed from the directory so no other program or user can access it. `du` shows that it is gone - it has walked the directory tree and the file is not listed. `df` shows that it is still there, as the filesystem knows that `more` is still using that space. Once you end the `more` session, `du` and `df` will agree.

Note that Soft Updates can delay the freeing of disk space; you might need to wait up to 30 seconds for the change to be visible!

This situation is common on web servers. Many people set up a FreeBSD web server and forget to rotate the log files. The access log fills up /var. The new administrator deletes the file, but the system still complains that the partition is full. Stopping and restarting the web server program would free the file, allowing the system to release the disk space. To prevent this from happening, set up newsyslog(8).

9.24. How can I add more swap space?

In the Configuration and Tuning section of the Handbook, you will find a section describing how to do this.

9.25. Why does FreeBSD see my disk as smaller than the manufacturer says it is?

Disk manufacturers calculate gigabytes as a billion bytes each, whereas FreeBSD calculates them as 1,073,741,824 bytes each. This explains why, for example, FreeBSD's boot messages will report a disk that supposedly has 80GB as holding 76319MB.

Also note that FreeBSD will (by default) reserve 8% of the disk space.

9.26. How is it possible for a partition to be more than 100% full?

A portion of each UFS partition (8%, by default) is reserved for use by the operating system and the `root` user. df(1) does not count that space when calculating the `Capacity` column, so it can exceed 100%. Also, you will notice that the `Blocks` column is always greater than the sum of the `Used` and

`Avail` columns, usually by a factor of 8%.

For more details, look up the `-m` option in tunefs(8).

Chapter 10. Διαχείριση Συστήματος

10.1. Where are the system start-up configuration files?

The primary configuration file is /etc/defaults/rc.conf (see rc.conf(5)) System startup scripts such as /etc/rc and /etc/rc.d (see rc(8)) just include this file. *Do not edit this file!* Instead, if there is any entry in /etc/defaults/rc.conf that you want to change, you should copy the line into /etc/rc.conf and change it there.

For example, if you wish to start named, the included DNS server, all you need to do is:

```
# echo named_enable="YES" >> /etc/rc.conf
```

To start up local services, place shell scripts in the /usr/local/etc/rc.d directory. These shell scripts should be set executable, and end with a .sh.

10.2. How do I add a user easily?

Use the adduser(8) command, or the pw(8) command for more complicated situations.

To remove the user, use the rmuser(8) command or, if necessary, pw(8).

10.3. Why do I keep getting messages like root: not found after editing my crontab file?

This is normally caused by editing the system crontab (/etc/crontab) and then using crontab(1) to install it:

```
# crontab /etc/crontab
```

This is not the correct way to do things. The system crontab has a different format to the per-user crontabs which crontab(1) updates (the crontab(5) manual page explains the differences in more detail).

If this is what you did, the extra crontab is simply a copy of /etc/crontab in the wrong format it. Delete it with the command:

```
# crontab -r
```

Next time, when you edit /etc/crontab, you should not do anything to inform cron(8) of the changes, since it will notice them automatically.

If you want something to be run once per day, week, or month, it is probably better to add shell scripts /usr/local/etc/periodic, and let the periodic(8) command run from the system cron schedule it with the other periodic system tasks.

The actual reason for the error is that the system crontab has an extra field, specifying which user to run the command as. In the default system crontab provided with FreeBSD, this is `root` for all entries. When this crontab is used as the `root` user's crontab (which is *not* the same as the system crontab), cron(8) assumes the string `root` is the first word of the command to execute, but no such command exists.

10.4. Why do I get the error, you are not in the correct group to su root when I try to su to root?

This is a security feature. In order to su to `root` (or any other account with superuser privileges), you must be in the `wheel` group. If this feature were not there, anybody with an account on a system who also found out `root`'s password would be able to gain superuser level access to the system. With this feature, this is not strictly true; su(1) will prevent them from even trying to enter the password if they are not in `wheel`.

To allow someone to su to `root`, simply put them in the `wheel` group.

10.5. I made a mistake in rc.conf, or another startup file, and now I cannot edit it because the filesystem is read-only. What should I do?

When you get the prompt to enter the shell pathname, simply press `ENTER`, and run `mount /` to re-mount the root filesystem in read/write mode. You may also need to run `mount -a -t ufs` to mount the filesystem where your favorite editor is defined. If your favorite editor is on a network filesystem, you will need to either configure the network manually before you can mount network filesystems, or use an editor which resides on a local filesystem, such as ed(1).

If you intend to use a full screen editor such as vi(1) or emacs(1), you may also need to run `export TERM=cons25` so that these editors can load the correct data from the termcap(5) database.

Once you have performed these steps, you can edit /etc/rc.conf as you usually would to fix the syntax error. The error message displayed immediately after the kernel boot messages should tell you the number of the line in the file which is at fault.

10.6. Why am I having trouble setting up my printer?

Please have a look at the Handbook entry on printing. It should cover most of your problem. See the Handbook entry on printing.

Some printers require a host-based driver to do any kind of printing. These so-called "WinPrinters" are not natively supported by FreeBSD. If your printer does not work in DOS or Windows NT® 4.0, it is probably a WinPrinter. Your only hope of getting one of these to work is to check if the

print/pnm2ppa port supports it.

10.7. How can I correct the keyboard mappings for my system?

Please see the Handbook section on using localization, specifically the section on console setup.

10.8. Why do I get messages like: unknown: PNP0303 can't assign resources on boot?

The following is an excerpt from a post to the freebsd-current mailing list.

Garrett Wollman <wollman@FreeBSD.org>, 24 April 2001 The "can't assign resources" messages indicate that the devices are legacy ISA devices for which a non-PnP-aware driver is compiled into the kernel. These include devices such as keyboard controllers, the programmable interrupt controller chip, and several other bits of standard infrastructure. The resources cannot be assigned because there is already a driver using those addresses.

10.9. Why can I not get user quotas to work properly?

1. It is possible that your kernel is not configured to use quotas. If this is the case, you will need to add the following line to your kernel configuration file and recompile:

   ```
   options QUOTA
   ```

   Please read theHandbook entry on quotas for full details.

2. Do not turn on quotas on /.
3. Put the quota file on the filesystem that the quotas are to be enforced on, i.e.:

Filesystem	Quota file
/usr	/usr/admin/quotas
/home	/home/admin/quotas
...	...

10.10. Does FreeBSD support System V IPC primitives?

Yes, FreeBSD supports System V-style IPC, including shared memory, messages and semaphores, in the GENERIC kernel. In a custom kernel, enable this support by adding the following lines to your kernel config.

```
options    SYSVSHM          # enable shared memory
options    SYSVSEM          # enable for semaphores
```

```
options    SYSVMSG         # enable for messaging
```

Recompile and install your kernel.

10.11. What other mail-server software can I use instead of Sendmail?

Sendmail is the default mail-server software for FreeBSD, but you can easily replace it with one of the other MTA (for instance, an MTA installed from the ports).

There are various alternative MTAs in the ports tree already, with mail/exim, mail/postfix, mail/qmail, and mail/zmailer being some of the most popular choices.

Diversity is nice, and the fact that you have many different mail-servers to chose from is considered a good thing; therefore try to avoid asking questions like "Is Sendmail better than Qmail?" in the mailing lists. If you do feel like asking, first check the mailing list archives. The advantages and disadvantages of each and every one of the available MTAs have already been discussed a few times.

10.12. I have forgotten the root password! What do I do?

Do not panic! Restart the system, type `boot -s` at the Boot: prompt to enter Single User mode. At the question about the shell to use, hit ENTER. You will be dropped to a # prompt. Enter `mount -u /` to remount your root filesystem read/write, then run `mount -a` to remount all the filesystems. Run `passwd root` to change the `root` password then run exit(1) to continue booting.

10.13. How do I keep ControlAltDelete from rebooting the system?

If you are using syscons (the default console driver) build and install a new kernel with the line:

```
options SC_DISABLE_REBOOT
```

in the configuration file. If you use the PCVT console driver, use the following kernel configuration line instead.

This can also be done by setting the following sysctl which does not require a reboot or kernel recompile:

```
# sysctl hw.syscons.kbd_reboot=0
```

```
options PCVT_CTRL_ALT_DEL
```

10.14. How do I reformat DOS text files to UNIX® ones?

Use this perl command:

```
% perl -i.bak -npe 's/\r\n/\n/g' file ...
```

file is the file(s) to process. The modification is done in-place, with the original file stored with a .bak extension.

Alternatively you can use the tr(1) command:

```
% tr -d '\r' < dos-text-file > unix-file
```

dos-text-file is the file containing DOS text while *unix-file* will contain the converted output. This can be quite a bit faster than using perl.

10.15. How do I kill processes by name?

Use killall(1).

10.16. Why is su bugging me about not being in root's ACL?

The error comes from the Kerberos distributed authentication system. The problem is not fatal but annoying. You can either run su with the -K option, or uninstall Kerberos as described in the next question.

10.17. How do I uninstall Kerberos?

To remove Kerberos from the system, reinstall the bin distribution for the release you are running. If you have the CDROM, you can mount the cd (we will assume on /cdrom) and run

```
# cd /cdrom/bin
# ./install.sh
```

Alternately, you can remove all `MAKE_KERBEROS` options from /etc/make.conf and rebuild world.

10.18. What happened to /dev/MAKEDEV?

FreeBSD 5.X and beyond use the devfs(8) device-on-demand system. Device drivers automatically create new device nodes as they are needed, obsoleting /dev/MAKEDEV.

If you are running FreeBSD 4.X or earlier and /dev/MAKEDEV is missing, then you really do have a problem. Grab a copy from the system source code, probably in /usr/src/etc/MAKEDEV.

10.19. How do I add pseudoterminals to the system?

If you have lots of telnet, ssh, X, or screen users, you will probably run out of pseudoterminals. Here is how to add more:

1. Build and install a new kernel with the line

   ```
   pseudo-device pty 256
   ```

   in the configuration file.

2. Run the commands

   ```
   # cd /dev
   # sh MAKEDEV pty{1,2,3,4,5,6,7}
   ```

   to make 256 device nodes for the new terminals.

3. Edit /etc/ttys and add lines for each of the 256 terminals. They should match the form of the existing entries, i.e. they look like

   ```
   ttyqc none network
   ```

   The order of the letter designations is `tty[pqrsPQRS][0-9a-v]`, using a regular expression.

4. Reboot the system with the new kernel and you are ready to go.

10.20. Why can I not create the snd0 device?

There is no snd device. The name is used as a shorthand for the various devices that make up the FreeBSD sound driver, such as mixer, sequencer, and dsp.

To create these devices you should

```
# cd /dev
```

```
# sh MAKEDEV snd0
```

You can omit this step if you are running FreeBSD 5.0-RELEASE or newer with devfs(5) enabled.

10.21. How do I re-read /etc/rc.conf and re-start /etc/rc without a reboot?

Go into single user mode and then back to multi user mode.

On the console do:

```
# shutdown now
(Note: without -r or -h)

# return
# exit
```

10.22. I tried to update my system to the latest -STABLE, but got -BETAx, -RC or -PRERELEASE! What is going on?

Short answer: it is just a name. RC stands for "Release Candidate". It signifies that a release is imminent. In FreeBSD, -PRERELEASE is typically synonymous with the code freeze before a release. (For some releases, the -BETA label was used in the same way as -PRERELEASE.)

Long answer: FreeBSD derives its releases from one of two places. Major, dot-zero, releases, such as 4.0-RELEASE and 5.0-RELEASE, are branched from the head of the development stream, commonly referred to as -CURRENT. Minor releases, such as 4.1-RELEASE or 5.2-RELEASE, have been snapshots of the active -STABLE branch. Starting with 4.3-RELEASE, each release also now has its own branch which can be tracked by people requiring an extremely conservative rate of development (typically only security advisories).

When a release is about to be made, the branch from which it will be derived from has to undergo a certain process. Part of this process is a code freeze. When a code freeze is initiated, the name of the branch is changed to reflect that it is about to become a release. For example, if the branch used to be called 4.5-STABLE, its name will be changed to 4.6-PRERELEASE to signify the code freeze and signify that extra pre-release testing should be happening. Bug fixes can still be committed to be part of the release. When the source code is in shape for the release the name will be changed to 4.6-RC to signify that a release is about to be made from it. Once in the RC stage, only the most critical bugs found can be fixed. Once the release (4.6-RELEASE in this example) and release branch have been made, the branch will be renamed to 4.6-STABLE.

For more information on version numbers and the various CVS branches, refer to the Release Engineering article.

10.23. I tried to install a new kernel, and the chflags failed. How do I get around this?

Short answer: You are probably at security level greater than 0. Reboot directly to single user mode to install the kernel.

Long answer: FreeBSD disallows changing system flags at security levels greater than 0. You can check your security level with the command:

```
# sysctl kern.securelevel
```

You cannot lower the security level; you have to boot to single mode to install the kernel, or change the security level in /etc/rc.conf then reboot. See the init(8) manual page for details on securelevel, and see /etc/defaults/rc.conf and the rc.conf(5) manual page for more information on rc.conf.

10.24. I cannot change the time on my system by more than one second! How do I get around this?

Short answer: You are probably at security level greater than 1. Reboot directly to single user mode to change the date.

Long answer: FreeBSD disallows changing the time by more that one second at security levels greater than 1. You can check your security level with the command:

```
# sysctl kern.securelevel
```

You cannot lower the security level; you have to boot to single mode to change the date, or change the security level in /etc/rc.conf then reboot. See the init(8) manual page for details on securelevel, and see /etc/defaults/rc.conf and the rc.conf(5) manual page for more information on rc.conf.

10.25. Why is rpc.statd using 256 megabytes of memory?

No, there is no memory leak, and it is not using 256 Mbytes of memory. For convenience, `rpc.statd` maps an obscene amount of memory into its address space. There is nothing terribly wrong with this from a technical standpoint; it just throws off things like top(1) and ps(1).

rpc.statd(8) maps its status file (resident on /var) into its address space; to save worrying about remapping it later when it needs to grow, it maps it with a generous size. This is very evident from the source code, where one can see that the length argument to mmap(2) is `0x10000000`, or one sixteenth of the address space on an IA32, or exactly 256MB.

10.26. Why can I not unset the schg file flag?

You are running at an elevated (i.e., greater than 0) securelevel. Lower the securelevel and try again. For more information, see the FAQ entry on securelevel and the init(8) manual page.

10.27. Why does SSH authentication through .shosts not work by default in recent versions of FreeBSD?

The reason why .shosts authentication does not work by default in more recent versions of FreeBSD is because ssh(1) is not installed suid `root` by default. To "fix" this, you can do one of the following:

- As a permanent fix, set `ENABLE_SUID_SSH` to `true` in /etc/make.conf and rebuild ssh (or run `make world`).
- As a temporary fix, change the mode on /usr/bin/ssh to `4555` by running `chmod 4555 /usr/bin/ssh` as `root`. Then add `ENABLE_SUID_SSH= true` to /etc/make.conf so the change takes effect the next time `make world` is run.

10.28. What is vnlru?

`vnlru` flushes and frees vnodes when the system hits the `kern.maxvnodes` limit. This kernel thread sits mostly idle, and only activates if you have a huge amount of RAM and are accessing tens of thousands of tiny files.

10.29. What do the various memory states displayed by top mean?

- `Active`: pages recently statistically used.
- `Inactive`: pages recently statistically unused.
- `Cache`: (most often) pages that have percolated from inactive to a status where they maintain their data, but can often be immediately reused (either with their old association, or reused with a new association.) There can be certain immediate transitions from `active` to `cache` state if the page is known to be clean (unmodified), but that transition is a matter of policy, depending upon the algorithm choice of the VM system maintainer.
- `Free`: pages without data content, and can be immediately used in certain circumstances where cache pages might be ineligible. Free pages can be reused at interrupt or process state.
- `Wired`: pages that are fixed into memory, usually for kernel purposes, but also sometimes for special use in processes.

Pages are most often written to disk (sort of a VM sync) when they are in the inactive state, but active pages can also be synced (but requires the availability of certain CPU features.) This depends upon the CPU tracking of the modified bit being available, and in certain situations there can be an advantage for a block of VM pages to be synced, whether they are active or inactive. In most common cases, it is best to think of the inactive queue to be a queue of relatively unused pages that might or might not be in the process of being written to disk. Cached pages are already synced, not

mapped, but available for immediate process use with their old association or with a new association. Free pages are available at interrupt level, but cached or free pages can be used at process state for reuse. Cache pages are not adequately locked to be available at interrupt level.

There are some other flags (e.g., busy flag or busy count) that might modify some of the rules that I described.

10.30. How much free memory is available?

There are a couple of kinds of "free memory". One kind is the amount of memory immediately available without paging anything else out. That is approximately the size of cache queue + size of free queue (with a derating factor, depending upon system tuning.) Another kind of "free memory" is the total amount of VM space. That can be complex, but is dependent upon the amount of swap space and memory. Other kinds of "free memory" descriptions are also possible, but it is relatively useless to define these, but rather it is important to make sure that the paging rate is kept low, and to avoid running out of swap space.

10.31. What is /var/empty? I can not delete it!

/var/empty is a directory that the sshd(8) program uses when performing privilege separation. The /var/empty directory is empty, owned by `root` and has the `schg` flag set.

Although it is not recommended to delete this directory, to do so you will need to unset the `schg` flag first. See the chflags(1) manual page for more information (and bear in mind the answer to the question on unsetting the schg flag).

Chapter 11. Το Σύστημα X Windows και οι Εικονικές Κονσόλες

11.1. What is the X Window System?

The X Window System (commonly `X11`) is the most widely available windowing system capable of running on UNIX® or UNIX® like systems, including FreeBSD. The X.Org Foundation administers the X protocol standards, with the current reference implementation, version 11 release 7.7, so you will often see references shortened to `X11`.

Many implementations are available for different architectures and operating systems. An implementation of the server-side code is properly known as an `X server`.

11.2. Which X implementations are available for FreeBSD?

Historically, the default implementation of X on FreeBSD has been XFree86™ which is maintained by The XFree86 Project, Inc. This software was installed by default on FreeBSD versions up until 4.10 and 5.2. Although Xorg itself maintained an implementation during that time period, it was basically only provided as a reference platform, as it had suffered greatly from bitrot over the years.

However, early in 2004, some XFree86 developers left that project over issues including the pace of code changes, future directions, and interpersonal conflicts, and are now contributing code directly to Xorg instead. At that time, Xorg updated its source tree to the last XFree86™ release before its subsequent licensing change (XFree86 version 4.3.99.903), incorporated many changes that had previously been maintained separately, and has released that software as X11R6.7.0. A separate but related project, freedesktop.org (or `fd.o` for short), is working on rearchitecting the original XFree86™ code to offload more work onto the graphics cards (with the goal of increased performance) and make it more modular (with the goal of increased maintainability, and thus faster releases as well as easier configuration). Xorg intends to incorporate the freedesktop.org changes in its future releases.

As of July 2004, in FreeBSD-CURRENT, XFree86™ has been replaced with Xorg as the default implementation. The XFree86™ ports (x11/XFree86-4 and subports) remain in the ports collection. But Xorg is the default X11 implementation for FreeBSD 5.3 and later.

For further information, read the X11 section of the FreeBSD Handbook.

The above describes the default X implementation installed. It is still possible to install either implementation by following the instructions in the entry for 20040723 in /usr/ports/UPDATING.

It is not currently possible to mix-and-match pieces of each implementation; one must choose one or the other.

11.3. Will my existing applications run with the Xorg suite?

The Xorg software is written to the same X11R6 specification that XFree86™ is, so basic applications should work unchanged. A few lesser-used protocols have been deprecated (`XIE`, `PEX`, and `lbxproxy`), but in the first two cases, the FreeBSD port of XFree86™ did not support them either.

11.4. Why did the X projects split, anyway?

The answer to this question is outside the scope of this FAQ. Note that there are voluminous postings in various mailing list archives on the Internet; please use your favorite search engine to investigate the history instead of asking this question on the FreeBSD mailing lists. It may even be the case that only the participants will ever know for certain.

11.5. Why did FreeBSD choose to go with the Xorg ports by default?

The Xorg developers claim that their goal is to release more often and incorporate new features more quickly. If they are able to do so, this will be very attractive. Also, their software still uses the traditional X license, while XFree86™ is now using their modified one.

> This decision is still controversial. Only time will tell which implementation proves technically superior. Each FreeBSD user should decide which they prefer.

11.6. I want to run X, how do I go about it?

If you would like to add X to an existing installation, you should use the x11/xorg meta-port, which will build and install all the necessary components.

Then read and follow the documentation on the xorgconfig(1) tool, which assists you in configuring Xorg for your particular graphics card/mouse/etc. You may also wish to examine the xorgcfg(1) tool, which provides a graphical interface to the X configuration process.

For further information, read the X11 section of the FreeBSD Handbook.

You may also wish to investigate the Xaccel server. See the section on Xi Graphics for more details.

11.7. I tried to run X, but I get an KDENABIO failed (Operation not permitted) error when I type startx. What do I do now?

Your system is probably running at a raised securelevel. It is not possible to start X at a raised securelevel because X requires write access to /dev/io. For more information, see at the init(8) manual page.

So the question is what else you should do instead, and you basically have two choices: set your securelevel back down to zero (usually from /etc/rc.conf), or run xdm(1) at boot time (before the securelevel is raised).

See How do I start XDM on boot? for more information about running xdm(1) at boot time.

11.8. Why does my mouse not work with X?

If you are using syscons (the default console driver), you can configure FreeBSD to support a mouse pointer on each virtual screen. In order to avoid conflicting with X, syscons supports a virtual device called /dev/sysmouse. All mouse events received from the real mouse device are written to the sysmouse device via moused. If you wish to use your mouse on one or more virtual consoles, *and* use X, see Είναι δυνατόν να χρησιμοποιηθεί το ποντίκι με κάποιο τρόπο έξω από το περιβάλλον του συστήματος X Window; and set up moused.

Then edit /etc/X11/xorg.conf and make sure you have the following lines:

```
Section "InputDevice"
   Option          "Protocol" "SysMouse"
   Option          "Device" "/dev/sysmouse"
.....
```

Some people prefer to use /dev/mouse under X. To make this work, /dev/mouse should be linked to /dev/sysmouse (see sysmouse(4)):

```
# cd /dev
# rm -f mouse
# ln -s sysmouse mouse
```

11.9. My mouse has a fancy wheel. Can I use it in X?

Yes.

You need to tell X that you have a 5 button mouse. To do this, simply add the lines `Buttons 5` and `ZAxisMapping 4 5` to the "InputDevice" section of /etc/X11/xorg.conf. For example, you might have the following "InputDevice" section in /etc/X11/xorg.conf.

Παράδειγμα 1. "InputDevice" Section for Wheeled Mouse in Xorg configuration file

```
Section "InputDevice"
   Identifier      "Mouse1"
   Driver          "mouse"
   Option          "Protocol" "auto"
   Option          "Device" "/dev/sysmouse"
   Option          "Buttons" "5"
   Option          "ZAxisMapping" "4 5"
```

```
EndSection
```

Παράδειγμα 2. ".emacs" example for naive page scrolling with Wheeled Mouse (optional)

```
;; wheel mouse
(global-set-key [mouse-4] 'scroll-down)
(global-set-key [mouse-5] 'scroll-up)
```

11.10. How do I use remote X displays?

For security reasons, the default setting is to not allow a machine to remotely open a window.

To enable this feature, simply start X with the optional `-listen_tcp` argument:

```
% startx -listen_tcp
```

11.11. Why do X Window menus and dialog boxes not work right?

Try turning off the `Num Lock` key.

If your `Num Lock` key is on by default at boot-time, you may add the following line in the `Keyboard` section of the /etc/X11/xorg.conf file.

```
# Let the server do the NumLock processing.  This should only be
# required when using pre-R6 clients
    ServerNumLock
```

11.12. What is a virtual console and how do I make more?

Virtual consoles, put simply, enable you to have several simultaneous sessions on the same machine without doing anything complicated like setting up a network or running X.

When the system starts, it will display a login prompt on the monitor after displaying all the boot messages. You can then type in your login name and password and start working (or playing!) on the first virtual console.

At some point, you will probably wish to start another session, perhaps to look at documentation for a program you are running or to read your mail while waiting for an FTP transfer to finish. Just do `Alt` + `F2` (hold down the `Alt` key and press the `F2` key), and you will find a login prompt waiting

for you on the second "virtual console"! When you want to go back to the original session, do `Alt` + `F1`.

The default FreeBSD installation has eight virtual consoles enabled. `Alt` + `F1`, `Alt` + `F2`, `Alt` + `F3`, and so on will switch between these virtual consoles.

To enable more of them, edit /etc/ttys (see ttys(5)) and add entries for ttyv4 to ttyvc after the comment on "Virtual terminals":

```
# Edit the existing entry for ttyv3 in /etc/ttys and change
# "off" to "on".
ttyv3   "/usr/libexec/getty Pc"         cons25  on secure
ttyv4   "/usr/libexec/getty Pc"         cons25  on secure
ttyv5   "/usr/libexec/getty Pc"         cons25  on secure
ttyv6   "/usr/libexec/getty Pc"         cons25  on secure
ttyv7   "/usr/libexec/getty Pc"         cons25  on secure
ttyv8   "/usr/libexec/getty Pc"         cons25  on secure
ttyv9   "/usr/libexec/getty Pc"         cons25  on secure
ttyva   "/usr/libexec/getty Pc"         cons25  on secure
ttyvb   "/usr/libexec/getty Pc"         cons25  on secure
```

Use as many or as few as you want. The more virtual terminals you have, the more resources that are used; this can be important if you have 8MB RAM or less. You may also want to change the `secure` to `insecure`.

> If you want to run an X server you *must* leave at least one virtual terminal unused (or turned off) for it to use. That is to say that if you want to have a login prompt pop up for all twelve of your Alt-function keys, you are out of luck - you can only do this for eleven of them if you also want to run an X server on the same machine.

The easiest way to disable a console is by turning it off. For example, if you had the full 12 terminal allocation mentioned above and you wanted to run X, you would change settings for virtual terminal 12 from:

```
ttyvb   "/usr/libexec/getty Pc"         cons25  on  secure
```

to:

```
ttyvb   "/usr/libexec/getty Pc"         cons25  off secure
```

If your keyboard has only ten function keys, you would end up with:

```
ttyv9   "/usr/libexec/getty Pc"         cons25  off secure
ttyva   "/usr/libexec/getty Pc"         cons25  off secure
```

```
ttyvb   "/usr/libexec/getty Pc"         cons25  off secure
```

(You could also just delete these lines.)

Next, the easiest (and cleanest) way to activate the virtual consoles is to reboot. However, if you really do not want to reboot, you can just shut down the X Window system and execute (as `root`):

```
# kill -HUP 1
```

It is imperative that you completely shut down X Window if it is running, before running this command. If you do not, your system will probably appear to hang/lock up after executing the kill command.

11.13. How do I access the virtual consoles from X?

Use `Ctrl` + `Alt` + `Fn` to switch back to a virtual console. `Ctrl` + `Alt` + `F1` would return you to the first virtual console.

Once you are back to a text console, you can then use `Alt` + `Fn` as normal to move between them.

To return to the X session, you must switch to the virtual console running X. If you invoked X from the command line, (e.g., using `startx`) then the X session will attach to the next unused virtual console, not the text console from which it was invoked. If you have eight active virtual terminals then X will be running on the ninth, and you would use `Alt` + `F9` to return.

11.14. How do I start XDM on boot?

There are two schools of thought on how to start xdm(1). One school starts xdm from /etc/ttys (see ttys(5)) using the supplied example, while the other simply runs xdm from rc.local (see rc(8)) or from a X.sh script in /usr/local/etc/rc.d. Both are equally valid, and one may work in situations where the other does not. In both cases the result is the same: X will pop up a graphical login: prompt.

The ttys method has the advantage of documenting which vty X will start on and passing the responsibility of restarting the X server on logout to init. The rc.local method makes it easy to kill xdm if there is a problem starting the X server.

If loaded from rc.local, `xdm` should be started without any arguments (i.e., as a daemon). xdm must start AFTER getty runs, or else getty and xdm will conflict, locking out the console. The best way around this is to have the script sleep 10 seconds or so then launch xdm.

If you are to start `xdm` from /etc/ttys, there still is a chance of conflict between `xdm` and getty(8). One way to avoid this is to add the `vt` number in the /usr/local/lib/X11/xdm/Xservers file.

```
:0 local /usr/local/bin/X vt4
```

The above example will direct the X server to run in /dev/ttyv3. Note the number is offset by one. The X server counts the vty from one, whereas the FreeBSD kernel numbers the vty from zero.

11.15. Why do I get Couldn't open console when I run xconsole?

If you start X with `startx`, the permissions on /dev/console will *not* get changed, resulting in things like `xterm -C` and `xconsole` not working.

This is because of the way console permissions are set by default. On a multi-user system, one does not necessarily want just any user to be able to write on the system console. For users who are logging directly onto a machine with a VTY, the fbtab(5) file exists to solve such problems.

In a nutshell, make sure an uncommented line of the form

```
/dev/ttyv0 0600 /dev/console
```

is in /etc/fbtab (see fbtab(5)) and it will ensure that whomever logs in on /dev/ttyv0 will own the console.

11.16. Before, I was able to run XFree86™ as a regular user. Why does it now say that I must be root?

All X servers need to be run as `root` in order to get direct access to your video hardware. Older versions of XFree86™ (= 3.3.6) installed all bundled servers to be automatically run as `root` (setuid to `root`). This is obviously a security hazard because X servers are large, complicated programs. Newer versions of XFree86™ do not install the servers setuid to `root` for just this reason.

Obviously, running an X server as the `root` user is not acceptable, nor a good idea security-wise. There are two ways to be able to use X as a regular user. The first is to use `xdm` or another display manager (e.g., `kdm`); the second is to use the `Xwrapper`.

`xdm` is a daemon that handles graphical logins. It is usually started at boot time, and is responsible for authenticating users and starting their sessions; it is essentially the graphical counterpart of getty(8) and login(1). For more information on `xdm` see the XFree86™ documentation, and the the FAQ entry on it.

`Xwrapper` is the X server wrapper; it is a small utility to enable one to manually run an X server while maintaining reasonable safety. It performs some sanity checks on the command line arguments given, and if they pass, runs the appropriate X server. If you do not want to run a display manager for whatever reason, this is for you. If you have installed the complete ports collection, you can find the port in /usr/ports/x11/wrapper.

11.17. Why does my PS/2 mouse misbehave under X?

Your mouse and the mouse driver may have somewhat become out of synchronization.

```
In rare cases the driver may erroneously report synchronization problem and you may
see the kernel message:
```

```
psmintr: out of sync (xxxx != yyyy)
```

and notice that your mouse does not work properly.

If this happens, disable the synchronization check code by setting the driver flags for the PS/2 mouse driver to 0x100. Enter *UserConfig* by giving the `-c` option at the boot prompt:

```
boot: -c
```

Then, in the *UserConfig* command line, type:

```
UserConfig> flags psm0 0x100
UserConfig> quit
```

11.18. Why does my PS/2 mouse from MouseSystems not work?

There have been some reports that certain model of PS/2 mouse from MouseSystems works only if it is put into the "high resolution" mode. Otherwise, the mouse cursor may jump to the upper-left corner of the screen every so often.

Specify the flags 0x04 to the PS/2 mouse driver to put the mouse into the high resolution mode. Enter *UserConfig* by giving the `-c` option at the boot prompt:

```
boot: -c
```

Then, in the *UserConfig* command line, type:

```
UserConfig> flags psm0 0x04
UserConfig> quit
```

See the previous section for another possible cause of mouse problems.

11.19. I want to install different X server.

FreeBSD versions prior 5.3 will use the default XFree86™ 4.X, while latter versions will default to Xorg. If you want to run a different X11 implementation than the default one, add the following line to /etc/make.conf, (if you do not have this file, create it):

```
X_WINDOW_SYSTEM=        xorg
```

This variable may be set to `xorg`, `xfree86-4`, or `xfree86-3`.

11.20. How do I reverse the mouse buttons?

Run the command `xmodmap -e "pointer = 3 2 1"` from your .xinitrc or .xsession.

11.21. How do I install a splash screen and where do I find them?

FreeBSD have a feature to allow the display of "splash" screens during the boot messages. The splash screens currently must be a 256 color bitmap (**.BMP**) or ZSoft PCX (.PCX) file. In addition, they must have a resolution of 320x200 or less to work on standard VGA adapters. If you compile VESA support into your kernel, then you can use larger bitmaps up to 1024x768. The actual VESA support can either be compiled directly into the kernel with the `VESA` kernel config option or by loading the VESA kld module during bootup.

To use a splash screen, you need to modify the startup files that control the boot process for FreeBSD.

You need to create a /boot/loader.rc file that contains the following lines:

```
include /boot/loader.4th
start
```

and a /boot/loader.conf that contains the following:

```
splash_bmp_load="YES"
bitmap_load="YES"
```

This assumes you are using /boot/splash.bmp for your splash screen. If you would rather use a PCX file, copy it to /boot/splash.pcx, create a /boot/loader.rc as instructed above, and create a /boot/loader.conf that contains:

```
splash_pcx_load="YES"
bitmap_load="YES"
bitmap_name="/boot/splash.pcx"
```

Now all you need is a splash screen. For that you can surf on over to the gallery at http://www.baldwin.cx/splash/.

11.22. Can I use the Windows® keys on my keyboard in X?

Yes. All you need to do is use xmodmap(1) to define what function you wish them to perform.

Assuming all "Windows®" keyboards are standard then the keycodes for the 3 keys are

- 115 - Windows® key, between the left-hand Ctrl and Alt keys
- 116 - Windows® key, to the right of the `AltGr` key
- 117 - `Menu` key, to the left of the right-hand `Ctrl` key

To have the left Windows® key print a comma, try this.

```
# xmodmap -e "keycode 115 = comma"
```

You will probably have to re-start your window manager to see the result.

To have the Windows® key-mappings enabled automatically every time you start X either put the `xmodmap` commands in your ~/.xinitrc file or, preferably, create a file ~/.xmodmaprc and include the `xmodmap` options, one per line, then add the line

```
xmodmap $HOME/.xmodmaprc
```

to your ~/.xinitrc.

For example, you could map the 3 keys to be `F13`, `F14`, and `F15`, respectively. This would make it easy to map them to useful functions within applications or your window manager, as demonstrated further down.

To do this put the following in ~/.xmodmaprc.

```
keycode 115 = F13
keycode 116 = F14
keycode 117 = F15
```

If you use `fvwm2`, for example, you could map the keys so that `F13` iconifies (or de-iconifies) the window the cursor is in, `F14` brings the window the cursor is in to the front or, if it is already at the front, pushes it to the back, and `F15` pops up the main Workplace (application) menu even if the cursor is not on the desktop, which is useful if you do not have any part of the desktop visible (and the logo on the key matches its functionality).

The following entries in ~/.fvwmrc implement the aforementioned setup:

```
Key F13        FTIWS    A        Iconify
Key F14        FTIWS    A        RaiseLower
```

```
Key F15        A        A        Menu Workplace Nop
```

11.23. How can I get 3D hardware acceleration for OpenGL®?

The availability of 3D acceleration depends on the version of XFree86™ or Xorg that you are using and the type of video chip you have. If you have an NVIDIA chip, you can use the binary drivers provided for FreeBSD on the Drivers section of their website. For other cards with XFree86™-4 or Xorg, including the Matrox G200/G400, ATI Rage 128/Radeon, and 3dfx Voodoo 3, 4, 5, and Banshee, information on hardware acceleration is available on the XFree86-4 Direct Rendering on FreeBSD page.

Chapter 12. Δικτύωση

12.1. Where can I get information on diskless booting?

"Diskless booting" means that the FreeBSD box is booted over a network, and reads the necessary files from a server instead of its hard disk. For full details, please read the Handbook entry on diskless booting

12.2. Can a FreeBSD box be used as a dedicated network router?

Yes. Please see the Handbook entry on advanced networking, specifically the section on routing and gateways.

12.3. Can I connect my Windows® box to the Internet via FreeBSD?

Typically, people who ask this question have two PCs at home, one with FreeBSD and one with some version of Windows® the idea is to use the FreeBSD box to connect to the Internet and then be able to access the Internet from the Windows® box through the FreeBSD box. This is really just a special case of the previous question and works perfectly well.

If you are using dialup to connect to the Internet user-mode ppp(8) contains a `-nat` option. If you run ppp(8) with the `-nat` option, set `gateway_enable` to *YES* in /etc/rc.conf, and configure your Windows® machine correctly, this should work fine. For more information, please see the ppp(8) manual page or the Handbook entry on user PPP.

If you are using kernel-mode PPP or have an Ethernet connection to the Internet, you need to use natd(8). Please look at the natd section of the Handbook for a tutorial.

12.4. Does FreeBSD support SLIP and PPP?

Yes. See the manual pages for slattach(8), sliplogin(8), ppp(8), and pppd(8). ppp(8) and pppd(8) provide support for both incoming and outgoing connections, while sliplogin(8) deals exclusively with incoming connections, and slattach(8) deals exclusively with outgoing connections.

For more information on how to use these, please see the Handbook chapter on PPP and SLIP.

If you only have access to the Internet through a "shell account", you may want to have a look at the net/slirp package. It can provide you with (limited) access to services such as ftp and http direct from your local machine.

12.5. Does FreeBSD support NAT or Masquerading?

Yes. If you want to use NAT over a user PPP connection, please see the Handbook entry on user PPP.

If you want to use NAT over some other sort of network connection, please look at the natd section of the Handbook.

12.6. How do I connect two FreeBSD systems over a parallel line using PLIP?

Please see the PLIP section of the Handbook.

12.7. Why can I not create a /dev/ed0 device?

Because they are not necessary. In the Berkeley networking framework, network interfaces are only directly accessible by kernel code. Please see the /etc/rc.network file and the manual pages for the various network programs mentioned there for more information. If this leaves you totally confused, then you should pick up a book describing network administration on another BSD-related operating system; with few significant exceptions, administering networking on FreeBSD is basically the same as on SunOS™ 4.0 or Ultrix.

12.8. How can I set up Ethernet aliases?

If the alias is on the same subnet as an address already configured on the interface, then add `netmask 0xffffffff` to your ifconfig(8) command-line, as in the following:

```
# ifconfig ed0 alias 192.0.2.2 netmask 0xffffffff
```

Otherwise, just specify the network address and netmask as usual:

```
# ifconfig ed0 alias 172.16.141.5 netmask 0xffffff00
```

12.9. How do I get my 3C503 to use the other network port?

If you want to use the other ports, you will have to specify an additional parameter on the ifconfig(8) command line. The default port is `link0`. To use the AUI port instead of the BNC one, use `link2`. These flags should be specified using the ifconfig_* variables in /etc/rc.conf (see rc.conf(5)).

12.10. Why am I having trouble with NFS and FreeBSD?

Certain PC network cards are better than others (to put it mildly) and can sometimes cause problems with network intensive applications like NFS.

See the Handbook entry on NFS for more information on this topic.

12.11. Why can I not NFS-mount from a Linux® box?

Some versions of the Linux® NFS code only accept mount requests from a privileged port; try

```
# mount -o -P linuxbox:/blah /mnt
```

12.12. Why can I not NFS-mount from a Sun box?

Sun™ workstations running SunOS™ 4.X only accept mount requests from a privileged port; try

```
# mount -o -P sunbox:/blah /mnt
```

12.13. Why does mountd keep telling me it can't change attributes and that I have a bad exports list on my FreeBSD NFS server?

The most frequent problem is not understanding the correct format of /etc/exports. Please review exports(5) and the NFS entry in the Handbook, especially the section on configuring NFS.

12.14. Why am I having problems talking PPP to NeXTStep machines?

Try disabling the TCP extensions in /etc/rc.conf (see rc.conf(5)) by changing the following variable to NO:

```
tcp_extensions=NO
```

Xylogic's Annex boxes are also broken in this regard and you must use the above change to connect through them.

12.15. How do I enable IP multicast support?

FreeBSD supports multicast host operations by default. If you want your box to run as a multicast router, you need to recompile your kernel with the `MROUTING` option and run mrouted(8). FreeBSD will start mrouted(8) at boot time if the flag `mrouted_enable` is set to `"YES"` in /etc/rc.conf.

MBONE tools are available in their own ports category, mbone. If you are looking for the conference tools `vic` and `vat`, look there!

12.16. Which network cards are based on the DEC PCI chipset?

Here is a list compiled by Glen Foster gfoster@driver.nsta.org, with some more modern additions:

Πίνακας 2. Network cards based on the DEC PCI chipset

Vendor	Model
ASUS	PCI-L101-TB
Accton	ENI1203
Cogent	EM960PCI
Compex	ENET32-PCI
D-Link	DE-530
Dayna	DP1203, DP2100
DEC	DE435, DE450
Danpex	EN-9400P3
JCIS	Condor JC1260
Linksys	EtherPCI
Mylex	LNP101
SMC	EtherPower 10/100 (Model 9332)
SMC	EtherPower (Model 8432)
TopWare	TE-3500P
Znyx (2.2.x)	ZX312, ZX314, ZX342, ZX345, ZX346, ZX348
Znyx (3.x)	ZX345Q, ZX346Q, ZX348Q, ZX412Q, ZX414, ZX442, ZX444, ZX474, ZX478, ZX212, ZX214 (10mbps/hd)

12.17. Why do I have to use the FQDN for hosts on my site?

You will probably find that the host is actually in a different domain; for example, if you are in foo.example.org and you wish to reach a host called `mumble` in the `example.org` domain, you will have to refer to it by the fully-qualified domain name, `mumble.example.org`, instead of just `mumble`.

Traditionally, this was allowed by BSD BIND resolvers. However the current version of bind (see named(8)) that ships with FreeBSD no longer provides default abbreviations for non-fully qualified domain names other than the domain you are in. So an unqualified host `mumble` must either be found as `mumble.foo.example.org`, or it will be searched for in the root domain.

This is different from the previous behavior, where the search continued across `mumble.example.org`, and `mumble.edu`. Have a look at RFC 1535 for why this was considered bad practice, or even a

security hole.

As a good workaround, you can place the line

```
search foo.example.org example.org
```

instead of the previous

```
domain foo.example.org
```

into your /etc/resolv.conf file (see resolv.conf(5)). However, make sure that the search order does not go beyond the "boundary between local and public administration", as RFC 1535 calls it.

12.18. Why do I get an error, Permission denied, for all networking operations?

If you have compiled your kernel with the `IPFIREWALL` option, you need to be aware that the default policy is to deny all packets that are not explicitly allowed.

If you had unintentionally misconfigured your system for firewalling, you can restore network operability by typing the following while logged in as `root`:

```
# ipfw add 65534 allow all from any to any
```

You can also set `firewall_type="open"` in /etc/rc.conf.

For further information on configuring a FreeBSD firewall, see the Handbook chapter.

12.19. How much overhead does IPFW incur?

Please see the Handbook's Firewalls section, specifically the section on IPFW Overhead Optimization.

12.20. Why is my ipfw fwd rule to redirect a service to another machine not working?

Possibly because you want to do network address translation (NAT) and not just forward packets. A "fwd" rule does exactly what it says; it forwards packets. It does not actually change the data inside the packet. Say we have a rule like:

```
01000 fwd`10.0.0.1`
from any to`foo 21`
```

When a packet with a destination address of *foo* arrives at the machine with this rule, the packet is forwarded to *10.0.0.1*, but it still has the destination address of *foo*! The destination address of the packet is *not* changed to *10.0.0.1*. Most machines would probably drop a packet that they receive with a destination address that is not their own. Therefore, using a "fwd" rule does not often work the way the user expects. This behavior is a feature and not a bug.

See the FAQ about redirecting services, the natd(8) manual, or one of the several port redirecting utilities in the ports collection for a correct way to do this.

12.21. How can I redirect service requests from one machine to another?

You can redirect FTP (and other service) request with the `socket` package, available in the ports tree in category "sysutils". Simply replace the service's command line to call socket instead, like so:

```
ftp stream tcp nowait nobody /usr/local/bin/socket socket ftp.example.com ftp
```

where *ftp.example.com* and *ftp* are the host and port to redirect to, respectively.

12.22. Where can I get a bandwidth management tool?

There are three bandwidth management tools available for FreeBSD. dummynet(4) is integrated into FreeBSD as part of ipfw(4). ALTQ is available for free on FreeBSD 4.X and has been integrated into FreeBSD 5.X as part of pf(4). Bandwidth Manager from Emerging Technologies is a commercial product.

12.23. Why do I get /dev/bpf0: device not configured?

You are running a program that requires the Berkeley Packet Filter (bpf(4)), but it is not in your kernel. Add this to your kernel config file and build a new kernel:

```
pseudo-device bpf        # Berkeley Packet Filter
```

On FreeBSD 4.X and earlier, you must also create the device node. After rebooting, go to the /dev directory and run:

```
# sh MAKEDEV bpf0
```

Please see the Handbook entry on device nodes for more information on managing devices.

12.24. How do I mount a disk from a Windows® machine that is on my network, like smbmount in Linux®?

Use the SMBFS toolset. It includes a set of kernel modifications and a set of userland programs. The programs and information are available as net/smbfs in the ports collection, or in the base system as of 4.5-RELEASE and later.

12.25. What are these messages about icmp-response bandwidth limit 300/200 pps in my log files?

This is the kernel telling you that some activity is provoking it to send more ICMP or TCP reset (RST) responses than it thinks it should. ICMP responses are often generated as a result of attempted connections to unused UDP ports. TCP resets are generated as a result of attempted connections to unopened TCP ports. Among others, these are the kinds of activities which may cause these messages:

- Brute-force denial of service (DoS) attacks (as opposed to single-packet attacks which exploit a specific vulnerability).
- Port scans which attempt to connect to a large number of ports (as opposed to only trying a few well-known ports).

The first number in the message tells you how many packets the kernel would have sent if the limit was not in place, and the second number tells you the limit. You can control the limit using the `net.inet.icmp.icmplim` sysctl variable like this, where `300` is the limit in packets per second:

```
# sysctl -w net.inet.icmp.icmplim=300
```

If you do not want to see messages about this in your log files, but you still want the kernel to do response limiting, you can use the `net.inet.icmp.icmplim_output` sysctl variable to disable the output like this:

```
# sysctl -w net.inet.icmp.icmplim_output=0
```

Finally, if you want to disable response limiting, you can set the `net.inet.icmp.icmplim` sysctl variable (see above for an example) to `0`. Disabling response limiting is discouraged for the reasons listed above.

12.26. What are these arp: unknown hardware address format error messages?

This means that some device on your local Ethernet is using a MAC address in a format that FreeBSD does not recognize. This is probably caused by someone experimenting with an Ethernet

card somewhere else on the network. You will see this most commonly on cable modem networks. It is harmless, and should not affect the performance of your FreeBSD machine.

12.27. I have just installed CVSup but trying to execute it produces errors. What is wrong?

First, see if the error message you are receiving is like the one shown below.

```
/usr/libexec/ld-elf.so.1: Shared object "libXaw.so.6" not found
```

Errors like these are caused by installing the net/cvsup port on a machine which does not have the XFree86™ suite. If you want to use the GUI included with CVSup you will need to install XFree86™ now. Alternatively if you just wish to use CVSup from a command line you should delete the package previously installed. Then install the net/cvsup-without-gui port. This is covered in more detail in the CVSup section of the Handbook.

Chapter 13. Ασφάλεια

13.1. What is a sandbox?

"Sandbox" is a security term. It can mean two things:

- A process which is placed inside a set of virtual walls that are designed to prevent someone who breaks into the process from being able to break into the wider system.

  The process is said to be able to "play" inside the walls. That is, nothing the process does in regards to executing code is supposed to be able to breech the walls so you do not have to do a detailed audit of its code to be able to say certain things about its security.

  The walls might be a userid, for example. This is the definition used in the security(7) and named(8) man pages.

  Take the `ntalk` service, for example (see /etc/inetd.conf). This service used to run as userid `root`. Now it runs as userid `tty`. The `tty` user is a sandbox designed to make it more difficult for someone who has successfully hacked into the system via ntalk from being able to hack beyond that user id.

- A process which is placed inside a simulation of the machine. This is more hard-core. Basically it means that someone who is able to break into the process may believe that he can break into the wider machine but is, in fact, only breaking into a simulation of that machine and not modifying any real data.

  The most common way to accomplish this is to build a simulated environment in a subdirectory and then run the processes in that directory chroot'd (i.e. / for that process is this directory, not the real / of the system).

  Another common use is to mount an underlying filesystem read-only and then create a filesystem layer on top of it that gives a process a seemingly writeable view into that filesystem. The process may believe it is able to write to those files, but only the process sees the effects - other processes in the system do not, necessarily.

  An attempt is made to make this sort of sandbox so transparent that the user (or hacker) does not realize that he is sitting in it.

UNIX® implements two core sandboxes. One is at the process level, and one is at the userid level.

Every UNIX® process is completely firewalled off from every other UNIX® process. One process cannot modify the address space of another. This is unlike Windows® where a process can easily overwrite the address space of any other, leading to a crash.

A UNIX® process is owned by a particular userid. If the userid is not the `root` user, it serves to firewall the process off from processes owned by other users. The userid is also used to firewall off on-disk data.

13.2. What is securelevel?

The securelevel is a security mechanism implemented in the kernel. Basically, when the securelevel is positive, the kernel restricts certain tasks; not even the superuser (i.e., `root`) is allowed to do them. At the time of this writing, the securelevel mechanism is capable of, among other things, limiting the ability to,

- unset certain file flags, such as `schg` (the system immutable flag),
- write to kernel memory via /dev/mem and /dev/kmem,
- load kernel modules, and
- alter firewall rules.

To check the status of the securelevel on a running system, simply execute the following command:

```
# sysctl kern.securelevel
```

The output will contain the name of the sysctl(8) variable (in this case, `kern.securelevel`) and a number. The latter is the current value of the securelevel. If it is positive (i.e., greater than 0), at least some of the securelevel's protections are enabled.

You cannot lower the securelevel of a running system; being able to do that would defeat its purpose. If you need to do a task that requires that the securelevel be non-positive (e.g., an `installworld` or changing the date), you will have to change the securelevel setting in /etc/rc.conf (you want to look for the `kern_securelevel` and `kern_securelevel_enable` variables) and reboot.

For more information on securelevel and the specific things all the levels do, please consult the init(8) manual page.

Securelevel is not a silver bullet; it has many known deficiencies. More often than not, it provides a false sense of security.

One of its biggest problems is that in order for it to be at all effective, all files used in the boot process up until the securelevel is set must be protected. If an attacker can get the system to execute their code prior to the securelevel being set (which happens quite late in the boot process since some things the system must do at start-up cannot be done at an elevated securelevel), its protections are invalidated. While this task of protecting all files used in the boot process is not technically impossible, if it is achieved, system maintenance will become a nightmare since one would have to take the system down, at least to single-user mode, to modify a configuration file.

This point and others are often discussed on the mailing lists, particularly the ηλεκτρονική λίστα του FreeBSD για θέματα ασφάλειας. Please search the archives here for an extensive discussion. Some people are hopeful that securelevel will soon go away in favor of a more fine-grained mechanism, but things are still hazy in this respect.

> Consider yourself warned.

13.3. BIND (named) is listening on port 53 and some other high-numbered port. What is going on?

BIND uses a random high-numbered port for outgoing queries. If you want to use port 53 for outgoing queries, either to get past a firewall or to make yourself feel better, you can try the following in /etc/namedb/named.conf:

```
options {
        query-source address * port 53;
};
```

You can replace the * with a single IP address if you want to tighten things further.

Congratulations, by the way. It is good practice to read your sockstat(1) output and notice odd things!

13.4. Sendmail is listening on port 587 as well as the standard port 25! What is going on?

Recent versions of Sendmail support a mail submission feature that runs over port 587. This is not yet widely supported, but is growing in popularity.

13.5. What is this UID 0 toor account? Have I been compromised?

Do not worry. `toor` is an "alternative" superuser account (toor is root spelt backwards). Previously it was created when the bash(1) shell was installed but now it is created by default. It is intended to be used with a non-standard shell so you do not have to change `root`'s default shell. This is important as shells which are not part of the base distribution (for example a shell installed from ports or packages) are likely to be installed in /usr/local/bin which, by default, resides on a different filesystem. If `root`'s shell is located in /usr/local/bin and /usr (or whatever filesystem contains /usr/local/bin) is not mounted for some reason, `root` will not be able to log in to fix a problem (although if you reboot into single user mode you will be prompted for the path to a shell).

Some people use `toor` for day-to-day `root` tasks with a non-standard shell, leaving `root`, with a standard shell, for single user mode or emergencies. By default you cannot log in using `toor` as it does not have a password, so log in as `root` and set a password for `toor` if you want to use it.

13.6. Why is suidperl not working properly?

For security reasons, `suidperl` is installed without the suid bit by default. The system administrator can enable suid behavior with the following command.

```
# chmod u+s /usr/bin/suidperl
```

If you want `suidperl` to be built suid during upgrades from source, edit /etc/make.conf and add `ENABLE_SUIDPERL=true` before you run `make buildworld`.

Chapter 14. PPP

14.1. I cannot make ppp(8) work. What am I doing wrong?

You should first read the ppp(8) manual page and the PPP section of the handbook. Enable logging with the command

```
set log Phase Chat Connect Carrier lcp ipcp ccp command
```

This command may be typed at the ppp(8) command prompt or it may be entered in the /etc/ppp/ppp.conf configuration file (the start of the `default` section is the best place to put it). Make sure that /etc/syslog.conf (see syslog.conf(5)) contains the lines

```
!ppp
*.*        /var/log/ppp.log
```

and that the file /var/log/ppp.log exists. You can now find out a lot about what is going on from the log file. Do not worry if it does not all make sense. If you need to get help from someone, it may make sense to them.

14.2. Why does ppp(8) hang when I run it?

This is usually because your hostname will not resolve. The best way to fix this is to make sure that /etc/hosts is consulted by your resolver first by editing /etc/host.conf and putting the `hosts` line first. Then, simply put an entry in /etc/hosts for your local machine. If you have no local network, change your `localhost` line:

```
127.0.0.1        foo.example.com foo localhost
```

Otherwise, simply add another entry for your host. Consult the relevant manual pages for more details.

You should be able to successfully `ping -c1 hostname` when you are done.

14.3. Why will ppp(8) not dial in -auto mode?

First, check that you have got a default route. By running `netstat -rn` (see netstat(1)), you should see two entries like this:

```
Destination        Gateway            Flags     Refs     Use     Netif Expire
default            10.0.0.2           UGSc        0        0      tun0
```

```
10.0.0.2          10.0.0.1          UH          0        0      tun0
```

This is assuming that you have used the addresses from the handbook, the manual page or from the ppp.conf.sample file. If you do not have a default route, it may be because you are running an old version of ppp(8) that does not understand the word `HISADDR` in the ppp.conf file.

Another reason for the default route line being missing is that you have mistakenly set up a default router in your /etc/rc.conf (see rc.conf(5)) file and you have omitted the line saying

```
delete ALL
```

from ppp.conf. If this is the case, go back to the Final system configuration section of the handbook.

14.4. What does No route to host mean?

This error is usually due to a missing

```
MYADDR:
  delete ALL
  add 0 0 HISADDR
```

section in your /etc/ppp/ppp.linkup file. This is only necessary if you have a dynamic IP address or do not know the address of your gateway. If you are using interactive mode, you can type the following after entering `packet mode` (packet mode is indicated by the capitalized PPP in the prompt):

```
delete ALL
add 0 0 HISADDR
```

Refer to the PPP and Dynamic IP addresses section of the handbook for further details.

14.5. Why does my connection drop after about 3 minutes?

The default PPP timeout is 3 minutes. This can be adjusted with the line

```
set timeout NNN
```

where *NNN* is the number of seconds of inactivity before the connection is closed. If *NNN* is zero, the connection is never closed due to a timeout. It is possible to put this command in the ppp.conf file, or to type it at the prompt in interactive mode. It is also possible to adjust it on the fly while the line is active by connecting to ppp's server socket using telnet(1) or pppctl(8). Refer to the ppp(8) man page for further details.

14.6. Why does my connection drop under heavy load?

If you have Link Quality Reporting (LQR) configured, it is possible that too many LQR packets are lost between your machine and the peer. Ppp deduces that the line must therefore be bad, and disconnects. Prior to FreeBSD version 2.2.5, LQR was enabled by default. It is now disabled by default. LQR can be disabled with the line

```
disable lqr
```

14.7. Why does my connection drop after a random amount of time?

Sometimes, on a noisy phone line or even on a line with call waiting enabled, your modem may hang up because it thinks (incorrectly) that it lost carrier.

There is a setting on most modems for determining how tolerant it should be to temporary losses of carrier. On a USR Sportster® for example, this is measured by the S10 register in tenths of a second. To make your modem more forgiving, you could add the following send-expect sequence to your dial string:

```
set dial "...... ATS10=10 OK ......"
```

Refer to your modem manual for details.

14.8. Why does my connection hang after a random amount of time?

Many people experience hung connections with no apparent explanation. The first thing to establish is which side of the link is hung.

If you are using an external modem, you can simply try using ping(8) to see if the TD light is flashing when you transmit data. If it flashes (and the RD light does not), the problem is with the remote end. If TD does not flash, the problem is local. With an internal modem, you will need to use the `set server` command in your ppp.conf file. When the hang occurs, connect to ppp(8) using pppctl(8). If your network connection suddenly revives (PPP was revived due to the activity on the diagnostic socket) or if you cannot connect (assuming the `set socket` command succeeded at startup time), the problem is local. If you can connect and things are still hung, enable local async logging with `set log local async` and use ping(8) from another window or terminal to make use of the link. The async logging will show you the data being transmitted and received on the link. If data is going out and not coming back, the problem is remote.

Having established whether the problem is local or remote, you now have two possibilities:

- If the problem is remote, read on entry The remote end is not responding. What can I do?.

- If the problem is local, read on entry ppp(8) has hung. What can I do?.

14.9. The remote end is not responding. What can I do?

There is very little you can do about this. Most ISPs will refuse to help if you are not running a Microsoft OS. You can `enable lqr` in your ppp.conf file, allowing ppp(8) to detect the remote failure and hang up, but this detection is relatively slow and therefore not that useful. You may want to avoid telling your ISP that you are running user-PPP...

First, try disabling all local compression by adding the following to your configuration:

```
disable pred1 deflate deflate24 protocomp acfcomp shortseq vj
deny pred1 deflate deflate24 protocomp acfcomp shortseq vj
```

Then reconnect to ensure that this makes no difference. If things improve or if the problem is solved completely, determine which setting makes the difference through trial and error. This will provide good ammunition when you contact your ISP (although it may make it apparent that you are not running a Microsoft product).

Before contacting your ISP, enable async logging locally and wait until the connection hangs again. This may use up quite a bit of disk space. The last data read from the port may be of interest. It is usually ascii data, and may even describe the problem ("Memory fault, core dumped"?).

If your ISP is helpful, they should be able to enable logging on their end, then when the next link drop occurs, they may be able to tell you why their side is having a problem. Feel free to send the details to Brian Somers <brian@FreeBSD.org>, or even to ask your ISP to contact me directly.

14.10. ppp(8) has hung. What can I do?

Your best bet here is to rebuild ppp(8) by adding `CFLAGS+=-g` and `STRIP=` to the end of the Makefile, then doing a `make clean make make install`. When ppp(8) hangs, find the ppp(8) process id with `ps ajxww | fgrep ppp` and run `gdb ppp PID`. From the gdb prompt, you can then use `bt` to get a stack trace.

Send the results to Brian Somers <brian@FreeBSD.org>.

14.11. Why does nothing happen after the Login OK! message?

Prior to FreeBSD version 2.2.5, once the link was established, ppp(8) would wait for the peer to initiate the Line Control Protocol (LCP). Many ISPs will not initiate negotiations and expect the client to do so. To force ppp(8) to initiate the LCP, use the following line:

```
set openmode active
```

It usually does no harm if both sides initiate negotiation, so openmode is now active by default. However, the next section explains when it *does* do some harm.

14.12. I keep seeing errors about magic being the same. What does it mean?

Occasionally, just after connecting, you may see messages in the log that say "magic is the same". Sometimes, these messages are harmless, and sometimes one side or the other exits. Most PPP implementations cannot survive this problem, and even if the link seems to come up, you will see repeated configure requests and configure acknowledgments in the log file until ppp(8) eventually gives up and closes the connection.

This normally happens on server machines with slow disks that are spawning a getty on the port, and executing ppp(8) from a login script or program after login. I have also heard reports of it happening consistently when using slirp. The reason is that in the time taken between getty(8) exiting and ppp(8) starting, the client-side ppp(8) starts sending Line Control Protocol (LCP) packets. Because ECHO is still switched on for the port on the server, the client ppp(8) sees these packets "reflect" back.

One part of the LCP negotiation is to establish a magic number for each side of the link so that "reflections" can be detected. The protocol says that when the peer tries to negotiate the same magic number, a NAK should be sent and a new magic number should be chosen. During the period that the server port has ECHO turned on, the client ppp(8) sends LCP packets, sees the same magic in the reflected packet and NAKs it. It also sees the NAK reflect (which also means ppp(8) must change its magic). This produces a potentially enormous number of magic number changes, all of which are happily piling into the server's tty buffer. As soon as ppp(8) starts on the server, it is flooded with magic number changes and almost immediately decides it has tried enough to negotiate LCP and gives up. Meanwhile, the client, who no longer sees the reflections, becomes happy just in time to see a hangup from the server.

This can be avoided by allowing the peer to start negotiating with the following line in your ppp.conf file:

```
set openmode passive
```

This tells ppp(8) to wait for the server to initiate LCP negotiations. Some servers however may never initiate negotiations. If this is the case, you can do something like:

```
set openmode active 3
```

This tells ppp(8) to be passive for 3 seconds, and then to start sending LCP requests. If the peer starts sending requests during this period, ppp(8) will immediately respond rather than waiting for the full 3 second period.

14.13. LCP negotiations continue until the connection is closed. What is wrong?

There is currently an implementation mis-feature in ppp(8) where it does not associate LCP, CCP IPCP responses with their original requests. As a result, if one PPP implementation is more than 6 seconds slower than the other side, the other side will send two additional LCP configuration requests. This is fatal.

Consider two implementations, `A` and `B`. `A` starts sending LCP requests immediately after connecting and `B` takes 7 seconds to start. When `B` starts, `A` has sent 3 LCP REQs. We are assuming the line has ECHO switched off, otherwise we would see magic number problems as described in the previous section. `B` sends a REQ, then an ACK to the first of `A`'s REQs. This results in `A` entering the OPENED state and sending and ACK (the first) back to `B`. In the meantime, `B` sends back two more ACKs in response to the two additional REQs sent by `A` before `B` started up. `B` then receives the first ACK from `A` and enters the OPENED state. `A` receives the second ACK from `B` and goes back to the REQ-SENT state, sending another (forth) REQ as per the RFC. It then receives the third ACK and enters the OPENED state. In the meantime, `B` receives the forth REQ from `A`, resulting in it reverting to the ACK-SENT state and sending another (second) REQ and (forth) ACK as per the RFC. `A` gets the REQ, goes into REQ-SENT and sends another REQ. It immediately receives the following ACK and enters OPENED.

This goes on until one side figures out that they are getting nowhere and gives up.

The best way to avoid this is to configure one side to be `passive` - that is, make one side wait for the other to start negotiating. This can be done with the

```
set openmode passive
```

command. Care should be taken with this option. You should also use the

```
set stopped N
```

command to limit the amount of time that ppp(8) waits for the peer to begin negotiations. Alternatively, the

```
set openmode active N
```

command (where *N* is the number of seconds to wait before starting negotiations) can be used. Check the manual page for details.

14.14. Why does ppp(8) lock up when I shell out to test it?

When you execute the `shell` or `!` command, ppp(8) executes a shell (or if you have passed any

arguments, ppp(8) will execute those arguments). Ppp will wait for the command to complete before continuing. If you attempt to use the PPP link while running the command, the link will appear to have frozen. This is because ppp(8) is waiting for the command to complete.

If you wish to execute commands like this, use the `!bg` command instead. This will execute the given command in the background, and ppp(8) can continue to service the link.

14.15. Why does ppp(8) over a null-modem cable never exit?

There is no way for ppp(8) to automatically determine that a direct connection has been dropped. This is due to the lines that are used in a null-modem serial cable. When using this sort of connection, LQR should always be enabled with the line

```
enable lqr
```

LQR is accepted by default if negotiated by the peer.

14.16. Why does ppp(8) dial for no reason in -auto mode?

If ppp(8) is dialing unexpectedly, you must determine the cause, and set up Dial filters (dfilters) to prevent such dialing.

To determine the cause, use the following line:

```
set log +tcp/ip
```

This will log all traffic through the connection. The next time the line comes up unexpectedly, you will see the reason logged with a convenient timestamp next to it.

You can now disable dialing under these circumstances. Usually, this sort of problem arises due to DNS lookups. To prevent DNS lookups from establishing a connection (this will *not* prevent ppp(8) from passing the packets through an established connection), use the following:

```
set dfilter 1 deny udp src eq 53
set dfilter 2 deny udp dst eq 53
set dfilter 3 permit 0/0 0/0
```

This is not always suitable, as it will effectively break your demand-dial capabilities - most programs will need a DNS lookup before doing any other network related things.

In the DNS case, you should try to determine what is actually trying to resolve a host name. A lot of the time, sendmail(8) is the culprit. You should make sure that you tell sendmail not to do any DNS

lookups in its configuration file. See the section on using email with a dialup connection in the FreeBSD Handbook for details on how to create your own configuration file and what should go into it. You may also want to add the following line to your .mc file:

```
define(`confDELIVERY_MODE', `d')dnl
```

This will make sendmail queue everything until the queue is run (usually, sendmail is invoked with `-bd -q30m`, telling it to run the queue every 30 minutes) or until a `sendmail -q` is done (perhaps from your ppp.linkup file).

14.17. What do these CCP errors mean?

I keep seeing the following errors in my log file:

```
CCP: CcpSendConfigReq
CCP: Received Terminate Ack (1) state = Req-Sent (6)
```

This is because ppp(8) is trying to negotiate Predictor1 compression, and the peer does not want to negotiate any compression at all. The messages are harmless, but if you wish to remove them, you can disable Predictor1 compression locally too:

```
disable pred1
```

14.18. Why does ppp(8) not log my connection speed?

In order to log all lines of your modem "conversation", you must enable the following:

```
set log +connect
```

This will make ppp(8) log everything up until the last requested "expect" string.

If you wish to see your connect speed and are using PAP or CHAP (and therefore do not have anything to "chat" after the CONNECT in the dial script - no `set login` script), you must make sure that you instruct ppp(8) to "expect" the whole CONNECT line, something like this:

```
set dial "ABORT BUSY ABORT NO\\sCARRIER TIMEOUT 4 \
  \"\" ATZ OK-ATZ-OK ATDT\\T TIMEOUT 60 CONNECT \\c \\n"
```

Here, we get our CONNECT, send nothing, then expect a line-feed, forcing ppp(8) to read the whole CONNECT response.

14.19. Why does ppp(8) ignore the \ character in my chat script?

Ppp parses each line in your config files so that it can interpret strings such as `set phone "123 456 789"` correctly and realize that the number is actually only *one* argument. In order to specify a `"` character, you must escape it using a backslash (`\`).

When the chat interpreter parses each argument, it re-interprets the argument in order to find any special escape sequences such as `\P` or `\T` (see the manual page). As a result of this double-parsing, you must remember to use the correct number of escapes.

If you wish to actually send a `\` character to (say) your modem, you would need something like:

```
set dial "\"\" ATZ OK-ATZ-OK AT\\\\X OK"
```

resulting in the following sequence:

```
ATZ
OK
AT\X
OK
```

or

```
set phone 1234567
set dial "\"\" ATZ OK ATDT\\T"
```

resulting in the following sequence:

```
ATZ
OK
ATDT1234567
```

14.20. Why does ppp(8) get a seg-fault, but I see no ppp.core file?

Ppp (or any other program for that matter) should never dump core. Because ppp(8) runs with an effective user id of 0, the operating system will not write ppp(8)'s core image to disk before terminating it. If, however ppp(8) is actually terminating due to a segmentation violation or some other signal that normally causes core to be dumped, *and* you are sure you are using the latest version (see the start of this section), then you should do the following:

```
% tar xfz ppp-*.src.tar.gz
```

```
% cd ppp*/ppp
% echo STRIP= >>Makefile
% echo CFLAGS+=-g >>Makefile
% make clean all
% su
# make install
# chmod 555 /usr/sbin/ppp
```

You will now have a debuggable version of ppp(8) installed. You will have to be `root` to run ppp(8) as all of its privileges have been revoked. When you start ppp(8), take a careful note of what your current directory was at the time.

Now, if and when ppp(8) receives the segmentation violation, it will dump a core file called ppp.core. You should then do the following:

```
% su
# gdb /usr/sbin/ppp ppp.core
(gdb) bt
..
(gdb) f 0
..
(gdb) i args
..
(gdb) l
..
```

All of this information should be given alongside your question, making it possible to diagnose the problem.

If you are familiar with gdb, you may wish to find out some other bits and pieces such as what actually caused the dump and the addresses values of the relevant variables.

14.21. Why does the process that forces a dial in auto mode never connect?

This was a known problem with ppp(8) set up to negotiate a dynamic local IP number with the peer in auto mode. It is fixed in the latest version - search the manual page for `iface`.

The problem was that when that initial program calls connect(2), the IP number of the tun interface is assigned to the socket endpoint. The kernel creates the first outgoing packet and writes it to the tun device. ppp(8) then reads the packet and establishes a connection. If, as a result of ppp(8)'s dynamic IP assignment, the interface address is changed, the original socket endpoint will be invalid. Any subsequent packets sent to the peer will usually be dropped. Even if they are not, any responses will not route back to the originating machine as the IP number is no longer owned by that machine.

There are several theoretical ways to approach this problem. It would be nicest if the peer would

re-assign the same IP number if possible `:-)` The current version of ppp(8) does this, but most other implementations do not.

The easiest method from our side would be to never change the tun interface IP number, but instead to change all outgoing packets so that the source IP number is changed from the interface IP to the negotiated IP on the fly. This is essentially what the `iface-alias` option in the latest version of ppp(8) is doing (with the help of libalias(3) and ppp(8)'s `-nat` switch) - it is maintaining all previous interface addresses and NATing them to the last negotiated address.

Another alternative (and probably the most reliable) would be to implement a system call that changes all bound sockets from one IP to another. ppp(8) would use this call to modify the sockets of all existing programs when a new IP number is negotiated. The same system call could be used by dhcp clients when they are forced to re-bind() their sockets.

Yet another possibility is to allow an interface to be brought up without an IP number. Outgoing packets would be given an IP number of 255.255.255.255 up until the first SIOCAIFADDR ioctl is done. This would result in fully binding the socket. It would be up to ppp(8) to change the source IP number, but only if it is set to 255.255.255.255, and only the IP number and IP checksum would need to change. This, however is a bit of a hack as the kernel would be sending bad packets to an improperly configured interface, on the assumption that some other mechanism is capable of fixing things retrospectively.

14.22. Why do most games not work with the -nat switch?

The reason games and the like do not work when libalias is in use is that the machine on the outside will try to open a connection or send (unsolicited) UDP packets to the machine on the inside. The NAT software does not know that it should send these packets to the interior machine.

To make things work, make sure that the only thing running is the software that you are having problems with, then either run tcpdump on the tun interface of the gateway or enable ppp(8) tcp/ip logging (`set log +tcp/ip`) on the gateway.

When you start the offending software, you should see packets passing through the gateway machine. When something comes back from the outside, it will be dropped (that is the problem). Note the port number of these packets then shut down the offending software. Do this a few times to see if the port numbers are consistent. If they are, then the following line in the relevant section of /etc/ppp/ppp.conf will make the software functional:

```
nat port proto internalmachine:port port
```

where *proto* is either `tcp` or `udp`, *internalmachine* is the machine that you want the packets to be sent to and *port* is the destination port number of the packets.

You will not be able to use the software on other machines without changing the above command, and running the software on two internal machines at the same time is out of the question - after all, the outside world is seeing your entire internal network as being just a single machine.

If the port numbers are not consistent, there are three more options:

1. Submit support in libalias. Examples of "special cases" can be found in /usr/src/lib/libalias/alias_*.c (alias_ftp.c is a good prototype). This usually involves reading certain recognised outgoing packets, identifying the instruction that tells the outside machine to initiate a connection back to the internal machine on a specific (random) port and setting up a "route" in the alias table so that the subsequent packets know where to go.

   This is the most difficult solution, but it is the best and will make the software work with multiple machines.

2. Use a proxy. The application may support socks5 for example, or (as in the "cvsup" case) may have a "passive" option that avoids ever requesting that the peer open connections back to the local machine.
3. Redirect everything to the internal machine using `nat addr`. This is the sledge-hammer approach.

14.23. Has anybody made a list of useful port numbers?

Not yet, but this is intended to grow into such a list (if any interest is shown). In each example, *internal* should be replaced with the IP number of the machine playing the game.

- Asheron's Call

  ```
  nat port udp internal :65000 65000
  ```

  Manually change the port number within the game to 65000. If you have got a number of machines that you wish to play on assign a unique port number for each (i.e. 65001, 65002, etc) and add a `nat port` line for each one.

- Half Life

  ```
  nat port udp internal:27005 27015
  ```

- PCAnywhere 8.0

  ```
  nat port udp internal:5632 5632
  ```

  ```
  nat port tcp internal:5631 5631
  ```

- Quake

  ```
  nat port udp internal:6112 6112
  ```

- Quake 2

  ```
  nat port udp internal:27901 27910
  ```

  ```
  nat port udp internal:60021 60021
  ```

  ```
  nat port udp internal:60040 60040
  ```

- Red Alert

  `nat port udp internal:8675 8675`

  `nat port udp internal:5009 5009`

14.24. What are FCS errors?

FCS stands for `F`rame `C`heck `S`equence. Each PPP packet has a checksum attached to ensure that the data being received is the data being sent. If the FCS of an incoming packet is incorrect, the packet is dropped and the HDLC FCS count is increased. The HDLC error values can be displayed using the `show hdlc` command.

If your link is bad (or if your serial driver is dropping packets), you will see the occasional FCS error. This is not usually worth worrying about although it does slow down the compression protocols substantially. If you have an external modem, make sure your cable is properly shielded from interference - this may eradicate the problem.

If your link freezes as soon as you have connected and you see a large number of FCS errors, this may be because your link is not 8 bit clean. Make sure your modem is not using software flow control (XON/XOFF). If your datalink *must* use software flow control, use the command `set accmap 0x000a0000` to tell ppp(8) to escape the `^Q` and `^S` characters.

Another reason for seeing too many FCS errors may be that the remote end has stopped talking PPP. You may want to enable `async` logging at this point to determine if the incoming data is actually a login or shell prompt. If you have a shell prompt at the remote end, it is possible to terminate ppp(8) without dropping the line by using the `close lcp` command (a following `term` command will reconnect you to the shell on the remote machine.

If nothing in your log file indicates why the link might have been terminated, you should ask the remote administrator (your ISP?) why the session was terminated.

14.25. Why do Mac OS® and Windows® 98 connections freeze when running PPPoE on the gateway?

Thanks to Michael Wozniak mwozniak@netcom.ca for figuring this out and Dan Flemming danflemming@mac.com for the Mac solution:

This is due to what is called a "Black Hole" router. Mac OS® and Windows® 98 (and maybe other Microsoft OSs) send TCP packets with a requested segment size too big to fit into a PPPoE frame (MTU is 1500 by default for Ethernet) *and* have the "do not fragment" bit set (default of TCP) and the Telco router is not sending ICMP "must fragment" back to the www site you are trying to load. (Alternatively, the router is sending the ICMP packet correctly, but the firewall at the www site is dropping it.) When the www server is sending you frames that do not fit into the PPPoE pipe the Telco router drops them on the floor and your page does not load (some pages/graphics do as they are smaller than a MSS.) This seems to be the default of most Telco PPPoE configurations (if only they knew how to program a router… sigh…)

One fix is to use regedit on your 95/98 boxes to add the following registry entry…

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Class\NetTrans\0000\MaxMTU
```

It should be a string with a value "1436", as some ADSL routers are reported to be unable to deal with packets larger than this. This registry key has been changed to `Tcpip\Parameters\Interfaces\ID for adapter\MTU` in Windows® 2000 and becomes a DWORD.

Refer to the Microsoft Knowledge Base documents Q158474 - Windows TCPIP Registry Entries and Q120642 - TCPIP NBT Configuration Parameters for Windows NT® for more information on changing Windows® MTU to work with a NAT router.

Another regedit possibility under Windows® 2000 is to set the `Tcpip\Parameters\Interfaces\ID for adapter\EnablePMTUBHDetect` DWORD to 1 as mentioned in the Microsoft document 120642 mentioned above.

Unfortunately, Mac OS® does not provide an interface for changing TCP/IP settings. However, there is commercial software available, such as OTAdvancedTuner (OT for OpenTransport, the Mac OS® TCP/IP stack) by Sustainable Softworks, that will allow users to customize TCP/IP settings. Mac OS® NAT users should select `ip_interface_MTU` from the drop-down menu, enter `1450` instead of `1500` in the box, click the box next to `Save as Auto Configure`, and click `Make Active`.

The latest version of ppp(8) (2.3 or greater) has an `enable tcpmssfixup` command that will automatically adjust the MSS to an appropriate value. This facility is enabled by default. If you are stuck with an older version of ppp(8), you may want to look at the tcpmssd port.

14.26. None of this helps - I am desperate! What can I do?

If all else fails, send as much information as you can, including your config files, how you are starting ppp(8), the relevant parts of your log file and the output of the `netstat -rn` command (before and after connecting) to the ηλεκτρονική λίστα γενικών ερωτήσεων του FreeBSD or the comp.unix.bsd.freebsd.misc news group, and someone should point you in the right direction.

Chapter 15. Σειριακές Επικοινωνίες

This section answers common questions about serial communications with FreeBSD. PPP and SLIP are covered in the Networking section.

15.1. How do I tell if FreeBSD found my serial ports?

As the FreeBSD kernel boots, it will probe for the serial ports in your system for which the kernel was configured. You can either watch your system closely for the messages it prints or run the command

```
% dmesg | grep sio
```

after your system is up and running.

Here is some example output from the above command:

```
sio0 at 0x3f8-0x3ff irq 4 on isa
sio0: type 16550A
sio1 at 0x2f8-0x2ff irq 3 on isa
sio1: type 16550A
```

This shows two serial ports. The first is on irq 4, is using port address `0x3f8`, and has a 16550A-type UART chip. The second uses the same kind of chip but is on irq 3 and is at port address `0x2f8`. Internal modem cards are treated just like serial ports---except that they always have a modem "attached" to the port.

The GENERIC kernel includes support for two serial ports using the same irq and port address settings in the above example. If these settings are not right for your system, or if you have added modem cards or have more serial ports than your kernel is configured for, just reconfigure your kernel. See section about building a kernel for more details.

15.2. How do I tell if FreeBSD found my modem cards?

Refer to the answer to the previous question.

15.3. How do I access the serial ports on FreeBSD?

The third serial port, sio2 (see sio(4), known as COM3 in DOS), is on /dev/cuaa2 for dial-out devices, and on /dev/ttyd2 for dial-in devices. What is the difference between these two classes of devices?

You use ttydX for dial-ins. When opening /dev/ttydX in blocking mode, a process will wait for the corresponding cuaaX device to become inactive, and then wait for the carrier detect line to go active. When you open the cuaaX device, it makes sure the serial port is not already in use by the ttydX device. If the port is available, it "steals" it from the ttydX device. Also, the cuaaX device does

not care about carrier detect. With this scheme and an auto-answer modem, you can have remote users log in and you can still dial out with the same modem and the system will take care of all the conflicts.

15.4. How do I enable support for a multiport serial card?

Again, the section on kernel configuration provides information about configuring your kernel. For a multiport serial card, place an sio(4) line for each serial port on the card in the kernel configuration file. But place the irq and vector specifiers on only one of the entries. All of the ports on the card should share one irq. For consistency, use the last serial port to specify the irq. Also, specify the `COM_MULTIPORT` option.

The following example is for an AST 4-port serial card on irq 7:

```
options "COM_MULTIPORT"
device sio4 at isa? port 0x2a0 tty flags 0x781
device sio5 at isa? port 0x2a8 tty flags 0x781
device sio6 at isa? port 0x2b0 tty flags 0x781
device sio7 at isa? port 0x2b8 tty flags 0x781 irq 7 vector siointr
```

The flags indicate that the master port has minor number 7 (`0x700`), diagnostics enabled during probe (`0x080`), and all the ports share an irq (`0x001`).

15.5. Can FreeBSD handle multiport serial cards sharing irqs?

Not yet. You will have to use a different irq for each card.

15.6. Can I set the default serial parameters for a port?

The ttydX (or cuaaX) device is the regular device you will want to open for your applications. When a process opens the device, it will have a default set of terminal I/O settings. You can see these settings with the command

```
# stty -a -f /dev/ttyd1
```

When you change the settings to this device, the settings are in effect until the device is closed. When it is reopened, it goes back to the default set. To make changes to the default set, you can open and adjust the settings of the "initial state" device. For example, to turn on CLOCAL mode, 8 bits, and XON/XOFF flow control by default for ttyd5, do:

```
# stty -f /dev/ttyid5 clocal cs8 ixon ixoff
```

A good place to do this is in /etc/rc.serial. Now, an application will have these settings by default when it opens ttyd5. It can still change these settings to its liking, though.

You can also prevent certain settings from being changed by an application by making adjustments to the "lock state" device. For example, to lock the speed of ttyd5 to 57600 bps, do

```
# stty -f /dev/ttyld5 57600
```

Now, an application that opens ttyd5 and tries to change the speed of the port will be stuck with 57600 bps.

Naturally, you should make the initial state and lock state devices writable only by `root`. The MAKEDEV(8) script does *NOT* do this when it creates the device entries.

15.7. How can I enable dialup logins on my modem?

So you want to become an Internet service provider, eh? First, you will need one or more modems that can auto-answer. Your modem will need to assert carrier-detect when it detects a carrier and not assert it all the time. It will need to hang up the phone and reset itself when the data terminal ready (DTR) line goes from on to off. It should probably use RTS/CTS flow control or no local flow control at all. Finally, it must use a constant speed between the computer and itself, but (to be nice to your callers) it should negotiate a speed between itself and the remote modem.

For many Hayes command-set—compatible modems, this command will make these settings and store them in nonvolatile memory:

```
AT C1 D3 K3 Q6 S0=1 W
```

See the section on sending AT commands below for information on how to make these settings without resorting to an MS-DOS® terminal program.

Next, make an entry in /etc/ttys (see ttys(5)) for the modem. This file lists all the ports on which the operating system will await logins. Add a line that looks something like this:

```
ttyd1 "/usr/libexec/getty std.57600" dialup on insecure
```

This line indicates that the second serial port (/dev/ttyd1) has a modem connected running at 57600 bps and no parity (`std.57600`, which comes from the file /etc/gettytab, see gettytab(5)). The terminal type for this port is `dialup`. The port is `on` and is `insecure`---meaning `root` logins on the port are not allowed. For dialin ports like this one, use the ttydX entry.

It is common practice to use `dialup` as the terminal type. Many users set up in their .profile or .login files a prompt for the actual terminal type if the starting type is dialup. The example shows the port as insecure. To become `root` on this port, you have to login as a regular user, then su(1) to become `root`. If you use `secure` then `root` can login in directly.

After making modifications to /etc/ttys, you need to send a hangup or HUP signal to the init(8) process:

```
# kill -HUP 1
```

This forces the init(8) process to reread /etc/ttys. The init process will then start getty processes on all `on` ports. You can find out if logins are available for your port by typing

```
% ps -ax | grep '[t]tyd1'
```

You should see something like:

```
747 ??  I      0:00.04 /usr/libexec/getty std.57600 ttyd1
```

15.8. How can I connect a dumb terminal to my FreeBSD box?

If you are using another computer as a terminal into your FreeBSD system, get a null-modem cable to go between the two serial ports. If you are using an actual terminal, see its accompanying instructions.

Then, modify /etc/ttys (see ttys(5)), like above. For example, if you are hooking up a WYSE-50 terminal to the fifth serial port, use an entry like this:

```
ttyd4 "/usr/libexec/getty std.38400" wyse50 on secure
```

This example shows that the port on /dev/ttyd4 has a wyse50 terminal connected at 38400 bps with no parity (`std.38400` from /etc/gettytab, see gettytab(5)) and `root` logins are allowed (`secure`).

15.9. Why can I not run tip or cu?

On your system, the programs tip(1) and cu(1) are probably executable only by `uucp` and group `dialer`. You can use the group `dialer` to control who has access to your modem or remote systems. Just add yourself to group dialer.

Alternatively, you can let everyone on your system run tip(1) and cu(1) by typing:

```
# chmod 4511 /usr/bin/cu
# chmod 4511 /usr/bin/tip
```

15.10. My stock Hayes modem is not supported---what can I do?

Actually, the manual page for tip(1) is out of date. There is a generic Hayes dialer already built in. Just use `at=hayes` in your /etc/remote (see remote(5)) file.

The Hayes driver is not smart enough to recognize some of the advanced features of newer modems---messages like `BUSY`, `NO DIALTONE`, or `CONNECT 115200` will just confuse it. You should turn those messages off when you use tip(1) (using `ATX0W`).

Also, the dial timeout for tip(1) is 60 seconds. Your modem should use something less, or else tip will think there is a communication problem. Try `ATS7=45W`.

Actually, as shipped tip(1) does not yet support it fully. The solution is to edit the file tipconf.h in the directory /usr/src/usr.bin/tip/tip. Obviously you need the source distribution to do this.

Edit the line `#define HAYES 0` to `#define HAYES 1`. Then `make` and `make install`. Everything works nicely after that.

15.11. How am I expected to enter these AT commands?

Make what is called a "direct" entry in your /etc/remote file (see remote(5)). For example, if your modem is hooked up to the first serial port, /dev/cuaa0, then put in the following line:

```
cuaa0:dv=/dev/cuaa0:br#19200:pa=none
```

Use the highest bps rate your modem supports in the br capability. Then, type `tip cuaa0` (see tip(1)) and you will be connected to your modem.

If there is no /dev/cuaa0 on your system, do this:

```
# cd /dev
# sh MAKEDEV cuaa0
```

Or use cu as `root` with the following command:

```
# cu -lline -sspeed
```

with *line* being the serial port (e.g. /dev/cuaa0) and *speed* being the speed (e.g.`57600`). When you are done entering the AT commands hit `~.` to exit.

15.12. Why does the @ sign for the pn capability not work?

The `@` sign in the phone number capability tells tip to look in /etc/phones for a phone number. But the `@` sign is also a special character in capability files like /etc/remote. Escape it with a backslash:

```
pn=\@
```

15.13. How can I dial a phone number on the command line?

Put what is called a "generic" entry in your /etc/remote file (see remote(5)). For example:

```
tip115200|Dial any phone number at 115200 bps:\
        :dv=/dev/cuaa0:br#115200:at=hayes:pa=none:du:
tip57600|Dial any phone number at 57600 bps:\
        :dv=/dev/cuaa0:br#57600:at=hayes:pa=none:du:
```

Then you can do something like `tip -115200 5551234`. If you prefer cu(1) over tip(1), use a generic cu entry:

```
cu115200|Use cu to dial any number at 115200bps:\
        :dv=/dev/cuaa1:br#57600:at=hayes:pa=none:du:
```

and type `cu 5551234 -s 115200`.

15.14. Do I have to type in the bps rate every time I do that?

Put in an entry for `tip1200` or `cu1200`, but go ahead and use whatever bps rate is appropriate with the br capability. tip(1) thinks a good default is 1200 bps which is why it looks for a `tip1200` entry. You do not have to use 1200 bps, though.

15.15. How can I more easily access a number of hosts through a terminal server?

Rather than waiting until you are connected and typing `CONNECT host` each time, use tip's `cm` capability. For example, these entries in /etc/remote (see remote(5)):

```
pain|pain.deep13.com|Forrester's machine:\
        :cm=CONNECT pain\n:tc=deep13:
```

```
muffin|muffin.deep13.com|Frank's machine:\
        :cm=CONNECT muffin\n:tc=deep13:
deep13:Gizmonics Institute terminal server:\
        :dv=/dev/cuaa2:br#38400:at=hayes:du:pa=none:pn=5551234:
```

will let you type `tip pain` or `tip muffin` to connect to the hosts `pain` or `muffin`; and `tip deep13` to get to the terminal server.

15.16. Can tip try more than one line for each site?

This is often a problem where a university has several modem lines and several thousand students trying to use them...

Make an entry for your university in /etc/remote (see remote(5)) and use `\@` for the `pn` capability:

```
big-university:\
        :pn=\@:tc=dialout
dialout:\
        :dv=/dev/cuaa3:br#9600:at=courier:du:pa=none:
```

Then, list the phone numbers for the university in /etc/phones (see phones(5)):

```
big-university 5551111
big-university 5551112
big-university 5551113
big-university 5551114
```

tip(1) will try each one in the listed order, then give up. If you want to keep retrying, run tip(1) in a while loop.

15.17. Why do I have to hit CTRLP twice to send CTRLP once?

`CTRL` + `P` is the default "force" character, used to tell tip(1) that the next character is literal data. You can set the force character to any other character with the `~s` escape, which means "set a variable".

Type `~sforce=single-char` followed by a newline. *single-char* is any single character. If you leave out *single-char*, then the force character is the nul character, which you can get by typing `CTRL` + `2` or `CTRL` + `SPACE`. A pretty good value for *single-char* is `SHIFT` + `CTRL` + `6`, which I have seen only used on some terminal servers.

You can have the force character be whatever you want by specifying the following in your $HOME/.tiprc file:

```
force=single-char
```

15.18. Why is everything I type suddenly in UPPER CASE?

You must have pressed `CTRL` + `A`, tip(1) "raise character", specially designed for people with broken `Caps Lock` keys. Use `~s` as above and set the variable "raisechar" to something reasonable. In fact, you can set it to the same as the force character, if you never expect to use either of these features.

Here is a sample .tiprc file perfect for Emacs users who need to type `CTRL` + `2` and `CTRL` + `A` a lot:

```
force=^^
raisechar=^^
```

The ^^ is `SHIFT` + `CTRL` + `6`.

15.19. How can I do file transfers with tip?

If you are talking to another UNIX® system, you can send and receive files with `~p` (put) and `~t` (take). These commands run cat(1) and echo(1) on the remote system to accept and send files. The syntax is:

```
~p local-file [remote-file]
~t remote-file [local-file]
```

There is no error checking, so you probably should use another protocol, like zmodem.

15.20. How can I run zmodem with tip?

First, install one of the zmodem programs from the ports collection (such as one of the two from the comms category, lrzsz or rzsz.

To receive files, start the sending program on the remote end. Then, press enter and type `~C rz` (or `~C lrz` if you installed lrzsz) to begin receiving them locally.

To send files, start the receiving program on the remote end. Then, press enter and type `~C sz files` (or `~C lsz files`) to send them to the remote system.

Chapter 16. Διάφορες Ερωτήσεις

16.1. FreeBSD uses far more swap space than Linux®. Why?

FreeBSD only appears to use more swap than Linux®. In actual fact, it does not. The main difference between FreeBSD and Linux® in this regard is that FreeBSD will proactively move entirely idle, unused pages of main memory into swap in order to make more main memory available for active use. Linux® tends to only move pages to swap as a last resort. The perceived heavier use of swap is balanced by the more efficient use of main memory.

Note that while FreeBSD is proactive in this regard, it does not arbitrarily decide to swap pages when the system is truly idle. Thus you will not find your system all paged out when you get up in the morning after leaving it idle overnight.

16.2. Why does top show very little free memory even when I have very few programs running?

The simple answer is that free memory is wasted memory. Any memory that your programs do not actively allocate is used within the FreeBSD kernel as disk cache. The values shown by top(1) labeled as `Inact`, `Cache`, and `Buf` are all cached data at different aging levels. This cached data means the system does not have to access a slow disk again for data it has accessed recently, thus increasing overall performance. In general, a low value shown for `Free` memory in top(1) is good, provided it is not *very* low.

16.3. Why will chmod not change the permissions on symlinks?

Symlinks do not have permissions, and by default, chmod(1) will not follow symlinks to change the permissions on the target file. So if you have a file, foo, and a symlink to that file, bar, then this command will always succeed.

```
% chmod g-w bar
```

However, the permissions on foo will not have changed.

You have to use either `-H` or `-L` together with the `-R` option to make this work. See the chmod(1) and symlink(7) manual pages for more info.

> The `-R` option does a *RECURSIVE* chmod(1). Be careful about specifying directories or symlinks to directories to chmod(1). If you want to change the permissions of a directory referenced by a symlink, use chmod(1) without any options and follow the symlink with a trailing slash (/). For example, if foo is a symlink to directory bar, and you want to change the permissions of foo (actually bar), you would do

something like:

```
% chmod 555 foo/
```

With the trailing slash, chmod(1) will follow the symlink, foo, to change the permissions of the directory, bar.

16.4. Can I run DOS binaries under FreeBSD?

Yes, you can use emulators/doscmd, a DOS emulation program, available in the FreeBSD Ports Collection.

> The doscmd program used to be an integrated part of FreeBSD, but was removed before the release of FreeBSD 5.3.

If doscmd will not suffice, the add-on utility emulators/pcemu emulates an 8088 and enough BIOS services to run many DOS text mode applications. It requires the X Window System.

16.5. What do I need to do to translate a FreeBSD document into my native language?

See the Translation FAQ in the FreeBSD Documentation Project Primer.

16.6. Why does my email to any address at FreeBSD.org bounce?

The FreeBSD.org mail system implements some of the stricter Postfix checks on incoming mail and rejects mail that is either misconfigured or is potential spam. Your mail might bounce for one of the following reasons:

- The email is being sent from a known spam domain or IP block.

  The FreeBSD mail servers reject email from known spam sources. If you have service through a company or domain who generates or relays spam, please switch to a service provider who does not.

- The body of the email only contains HTML.

  Mail should be sent in plain text only. Please configure your mail user agent to send plain text.

- The mailer at FreeBSD.org cannot resolve the IP address of the connecting host back to a symbolic name.

  Working reverse DNS is a standard requirement for accepting mail from a host. Set up reverse DNS for your mail server's IP address. Many home services (DSL, cable, dialup, etc.) will not give you this option. In this case, relay your email through your service provider's mail server.

- The hostname given in the EHLO/HELO part of the SMTP exchange cannot be resolved to an IP address.

  A fully qualified, resolvable host name is necessary in this part of the SMTP dialogue before mail will be accepted. If you do not have a host name that is registered in the DNS, then you should use your service provider's mail server to relay your mail.

- Your message had a message ID ending with the string "localhost".

  Some mail user agents generate bad message IDs which will not be accepted. You will need to persuade your mail user agent to generate a valid message ID or else configure your mail transfer agent to rewrite them.

16.7. Where can I find a free FreeBSD account?

While FreeBSD does not provide open access to any of their servers, others do provide open access UNIX® systems. The charge varies and limited services may be available.

Arbornet, Inc, also known as M-Net, has been providing open access to UNIX® systems since 1983. Starting on an Altos running System III, the site switched to BSD/OS in 1991. In June of 2000, the site switched again to FreeBSD. M-Net can be accessed via telnet and SSH and provides basic access to the entire FreeBSD software suite. However, network access is limited to members and patrons who donate to the system, which is run as a non-profit organization. M-Net also provides an bulletin board system and interactive chat.

Grex provides a site very similar to M-Net including the same bulletin board and interactive chat software. However, the machine is a Sun™ 4M and is running SunOS™.

16.8. What is sup, and how do I use it?

SUP stands for Software Update Protocol, and was developed by CMU for keeping their development trees in sync. We used it to keep remote sites in sync with our central development sources.

SUP is not bandwidth friendly, and has been retired. The current recommended method to keep your sources up to date is CVSup

16.9. What is the cute little red guy's name?

He does not have one, and is just called "the BSD daemon". If you insist upon using a name, call him "beastie". Note that "beastie" is pronounced "BSD".

You can learn more about the BSD daemon on his home page.

16.10. Can I use the BSD daemon image?

Perhaps. The BSD daemon is copyrighted by Marshall Kirk McKusick. You will want to check his Statement on the Use of the BSD Daemon Figure for detailed usage terms.

In summary, you are free to use the image in a tasteful manner, for personal use, so long as appropriate credit is given. If you want to use him commercially, you must contact Kirk McKusick. More details are available on the BSD Daemon's home page.

16.11. Do you have any BSD daemon images I could use?

You will find eps and Xfig drawings under /usr/shared/examples/BSD_daemon/.

16.12. I have seen an acronym or other term on the mailing lists and I do not understand what it means. Where should I look?

Please see the FreeBSD Glossary.

16.13. Why should I care what color the bikeshed is?

The really, really short answer is that you should not. The somewhat longer answer is that just because you are capable of building a bikeshed does not mean you should stop others from building one just because you do not like the color they plan to paint it. This is a metaphor indicating that you need not argue about every little feature just because you know enough to do so. Some people have commented that the amount of noise generated by a change is inversely proportional to the complexity of the change.

The longer and more complete answer is that after a very long argument about whether sleep(1) should take fractional second arguments, Poul-Henning Kamp <phk@FreeBSD.org> posted a long message entitled "A bike shed (any color will do) on greener grass...". The appropriate portions of that message are quoted below.

Poul-Henning Kamp <phk@FreeBSD.org> on freebsd-hackers, October 2, 1999 "What is it about this bike shed?" Some of you have asked me.

It is a long story, or rather it is an old story, but it is quite short actually. C. Northcote Parkinson wrote a book in the early 1960s, called "Parkinson's Law", which contains a lot of insight into the dynamics of management.

In the specific example involving the bike shed, the other vital component is an atomic power-plant, I guess that illustrates the age of the book.

Parkinson shows how you can go into the board of directors and get approval for building a multi-million or even billion dollar atomic power plant, but if you want to build a bike shed you will be tangled up in endless discussions.

Parkinson explains that this is because an atomic plant is so vast, so expensive and so complicated that people cannot grasp it, and rather than try, they fall back on the assumption that somebody else checked all the details before it got this far. Richard P. Feynmann gives a couple of interesting,

and very much to the point, examples relating to Los Alamos in his books.

A bike shed on the other hand. Anyone can build one of those over a weekend, and still have time to watch the game on TV. So no matter how well prepared, no matter how reasonable you are with your proposal, somebody will seize the chance to show that he is doing his job, that he is paying attention, that he is *here*.

In Denmark we call it "setting your fingerprint". It is about personal pride and prestige, it is about being able to point somewhere and say "There! *I* did that." It is a strong trait in politicians, but present in most people given the chance. Just think about footsteps in wet cement.

Chapter 17. Χιούμορ και FreeBSD

17.1. How cool is FreeBSD?

1. *Has anyone done any temperature testing while running FreeBSD? I know Linux® runs cooler than DOS, but have never seen a mention of FreeBSD. It seems to run really hot.*

   No, but we have done numerous taste tests on blindfolded volunteers who have also had 250 micrograms of LSD-25 administered beforehand. 35% of the volunteers said that FreeBSD tasted sort of orange, whereas Linux® tasted like purple haze. Neither group mentioned any significant variances in temperature. We eventually had to throw the results of this survey out entirely anyway when we found that too many volunteers were wandering out of the room during the tests, thus skewing the results. We think most of the volunteers are at Apple now, working on their new "scratch and sniff" GUI. It is a funny old business we are in!

Seriously, both FreeBSD and Linux® use the HLT (halt) instruction when the system is idle thus lowering its energy consumption and therefore the heat it generates. Also if you have APM (advanced power management) configured, then FreeBSD can also put the CPU into a low power mode.

17.2. Who is scratching in my memory banks??

1. *Is there anything "odd" that FreeBSD does when compiling the kernel which would cause the memory to make a scratchy sound? When compiling (and for a brief moment after recognizing the floppy drive upon startup, as well), a strange scratchy sound emanates from what appears to be the memory banks.*

   Yes! You will see frequent references to "daemons" in the BSD documentation, and what most people do not know is that this refers to genuine, non-corporeal entities that now possess your computer. The scratchy sound coming from your memory is actually high-pitched whispering exchanged among the daemons as they best decide how to deal with various system administration tasks.

If the noise gets to you, a good `fdisk /mbr` from DOS will get rid of them, but do not be surprised if they react adversely and try to stop you. In fact, if at any point during the exercise you hear the satanic voice of Bill Gates coming from the built-in speaker, take off running and do not ever look back! Freed from the counterbalancing influence of the BSD daemons, the twin demons of DOS and Windows® are often able to re-assert total control over your machine to the eternal damnation of your soul. Now that you know, given a choice you would probably prefer to get used to the scratchy noises, no?

17.3. How many FreeBSD hackers does it take to change a lightbulb?

One thousand, one hundred and sixty-nine:

Twenty-three to complain to -CURRENT about the lights being out;

Four to claim that it is a configuration problem, and that such matters really belong on -questions;

Three to submit PRs about it, one of which is misfiled under doc and consists only of "it's dark";

One to commit an untested lightbulb which breaks buildworld, then back it out five minutes later;

Eight to flame the PR originators for not including patches in their PRs;

Five to complain about buildworld being broken;

Thirty-one to answer that it works for them, and they must have cvsupped at a bad time;

One to post a patch for a new lightbulb to -hackers;

One to complain that he had patches for this three years ago, but when he sent them to -CURRENT they were just ignored, and he has had bad experiences with the PR system; besides, the proposed new lightbulb is non-reflexive;

Thirty-seven to scream that lightbulbs do not belong in the base system, that committers have no right to do things like this without consulting the Community, and WHAT IS -CORE DOING ABOUT IT!?

Two hundred to complain about the color of the bicycle shed;

Three to point out that the patch breaks style(9);

Seventeen to complain that the proposed new lightbulb is under GPL;

Five hundred and eighty-six to engage in a flame war about the comparative advantages of the GPL, the BSD license, the MIT license, the NPL, and the personal hygiene of unnamed FSF founders;

Seven to move various portions of the thread to -chat and -advocacy;

One to commit the suggested lightbulb, even though it shines dimmer than the old one;

Two to back it out with a furious flame of a commit message, arguing that FreeBSD is better off in the dark than with a dim lightbulb;

Forty-six to argue vociferously about the backing out of the dim lightbulb and demanding a statement from -core;

Eleven to request a smaller lightbulb so it will fit their Tamagotchi if we ever decide to port FreeBSD to that platform;

Seventy-three to complain about the SNR on -hackers and -chat and unsubscribe in protest;

Thirteen to post "unsubscribe", "How do I unsubscribe?", or "Please remove me from the list", followed by the usual footer;

One to commit a working lightbulb while everybody is too busy flaming everybody else to notice;

Thirty-one to point out that the new lightbulb would shine 0.364% brighter if compiled with TenDRA (although it will have to be reshaped into a cube), and that FreeBSD should therefore

switch to TenDRA instead of GCC;

One to complain that the new lightbulb lacks fairings;

Nine (including the PR originators) to ask "what is MFC?";

Fifty-seven to complain about the lights being out two weeks after the bulb has been changed.

Nik Clayton <nik@FreeBSD.org> adds:

I was laughing quite hard at this.

And then I thought, "Hang on, shouldn't there be '1 to document it.' in that list somewhere?"

And then I was enlightened :-)

17.4. Where does data written to /dev/null go?

It goes into a special data sink in the CPU where it is converted to heat which is vented through the heatsink / fan assembly. This is why CPU cooling is increasingly important; as people get used to faster processors, they become careless with their data and more and more of it ends up in /dev/null, overheating their CPUs. If you delete /dev/null (which effectively disables the CPU data sink) your CPU may run cooler but your system will quickly become constipated with all that excess data and start to behave erratically. If you have a fast network connection you can cool down your CPU by reading data out of /dev/random and sending it off somewhere; however you run the risk of overheating your network connection and / or angering your ISP, as most of the data will end up getting converted to heat by their equipment, but they generally have good cooling, so if you do not overdo it you should be OK.

Paul Robinson adds:

There are other methods. As every good sysadmin knows, it is part of standard practice to send data to the screen of interesting variety to keep all the pixies that make up your picture happy. Screen pixies (commonly mis-typed or re-named as "pixels" are categorized by the type of hat they wear (red, green or blue) and will hide or appear (thereby showing the color of their hat) whenever they receive a little piece of food. Video cards turn data into pixie-food, and then send them to the pixies - the more expensive the card, the better the food, so the better behaved the pixies are. They also need constant stimulation - this is why screen savers exist.

To take your suggestions further, you could just throw the random data to console, thereby letting the pixies consume it. This causes no heat to be produced at all, keeps the pixies happy and gets rid of your data quite quickly, even if it does make things look a bit messy on your screen.

Incidentally, as an ex-admin of a large ISP who experienced many problems attempting to maintain a stable temperature in a server room, I would strongly discourage people sending the data they do not want out to the network. The fairies who do the packet switching and routing get annoyed by it as well.

Chapter 18. Προχωρημένα Θέματα

18.1. How can I learn more about FreeBSD's internals?

At this time, there is only one book on FreeBSD-specific OS internals, namely "The Design and Implementation of the FreeBSD Operating System" by Marshall Kirk McKusick and George V. Neville-Neil, ISBN 0-201-70245-2, which focuses on version 5.X of FreeBSD.

Additionally, much general UNIX® knowledge is directly applicable to FreeBSD.

For a list of relevant books, please check the Handbook's Operating System Internals Bibliography.

18.2. How can I contribute to FreeBSD?

Please see the article on Contributing to FreeBSD for specific advice on how to do this. Assistance is more than welcome!

18.3. What are SNAPs and RELEASEs?

There are currently three active/semi-active branches in the FreeBSD CVS Repository. (Earlier branches are only changed very rarely, which is why there are only three active branches of development):

- `RELENG_5` AKA *5-STABLE*
- `RELENG_6` AKA *6-STABLE*
- `HEAD` AKA *-CURRENT* AKA *7.X-CURRENT*

`HEAD` is not an actual branch tag, like the other two; it is simply a symbolic constant for "*the current, non-branched development stream*" which we simply refer to as "-CURRENT".

Right now, "-CURRENT" is the 7.X development stream; the `5-STABLE` branch, RELENG_5, forked off from "-CURRENT" in October 2004, and the `6-STABLE` branch, RELENG_6, forked off from "-CURRENT" in November 2005.

18.4. How do I make my own custom release?

Please see the Release Engineering article.

18.5. Why does make world clobber my existing installed binaries?

Yes, this is the general idea; as its name might suggest, `make world` rebuilds every system binary from scratch, so you can be certain of having a clean and consistent environment at the end (which is why it takes so long).

If the environment variable `DESTDIR` is defined while running `make world` or `make install`, the newly-created binaries will be deposited in a directory tree identical to the installed one, rooted at `${DESTDIR}`. Some random combination of shared libraries modifications and program rebuilds can cause this to fail in `make world` however.

18.6. Why isn't cvsup.FreeBSD.org a round robin DNS entry to share the load amongst the various CVSup servers?

While CVSup mirrors update from the master CVSup server hourly, this update might happen at any time during the hour. This means that some servers have newer code than others, even though all servers have code that is less than an hour old. If `cvsup.FreeBSD.org` was a round robin DNS entry that simply redirected users to a random CVSup server, running CVSup twice in a row could download code older than the code already on the system.

18.7. Why does my system say (bus speed defaulted) when it boots?

The Adaptec 1542 SCSI host adapters allow the user to configure their bus access speed in software. Previous versions of the 1542 driver tried to determine the fastest usable speed and set the adapter to that. We found that this breaks some users' systems, so you now have to define the TUNE_1542 kernel configuration option in order to have this take place. Using it on those systems where it works may make your disks run faster, but on those systems where it does not, your data could be corrupted.

18.8. Can I follow -CURRENT with limited Internet access?

Yes, you can do this *without* downloading the whole source tree by using the CTM facility.

18.9. How did you split the distribution into 240k files?

Newer BSD based systems have a `-b` option to split(1) that allows them to split files on arbitrary byte boundaries.

Here is an example from /usr/src/Makefile.

```
bin-tarball:
(cd ${DISTDIR}; \
tar cf - . \
gzip --no-name -9 -c | \
split -b 240640 - \
${RELEASEDIR}/tarballs/bindist/bin_tgz.)
```

18.10. I have written a kernel extension, who do I send it to?

Please take a look at the article on Contributing to FreeBSD to learn how to submit code.

And thanks for the thought!

18.11. How are Plug N Play ISA cards detected and initialized?

By: Frank Durda IV uhclem@nemesis.lonestar.org

In a nutshell, there a few I/O ports that all of the PnP boards respond to when the host asks if anyone is out there. So when the PnP probe routine starts, it asks if there are any PnP boards present, and all the PnP boards respond with their model # to a I/O read of the same port, so the probe routine gets a wired-OR "yes" to that question. At least one bit will be on in that reply. Then the probe code is able to cause boards with board model IDs (assigned by Microsoft/Intel) lower than X to go "off-line". It then looks to see if any boards are still responding to the query. If the answer was `0`, then there are no boards with IDs above X. Now probe asks if there are any boards below `X`. If so, probe knows there are boards with a model numbers below X. Probe then asks for boards greater than X-(limit/4) to go off-line. If repeats the query. By repeating this semi-binary search of IDs-in-range enough times, the probing code will eventually identify all PnP boards present in a given machine with a number of iterations that is much lower than what 2^64 would take.

The IDs are two 32-bit fields (hence 2^64) + 8 bit checksum. The first 32 bits are a vendor identifier. They never come out and say it, but it appears to be assumed that different types of boards from the same vendor could have different 32-bit vendor ids. The idea of needing 32 bits just for unique manufacturers is a bit excessive.

The lower 32 bits are a serial #, Ethernet address, something that makes this one board unique. The vendor must never produce a second board that has the same lower 32 bits unless the upper 32 bits are also different. So you can have multiple boards of the same type in the machine and the full 64 bits will still be unique.

The 32 bit groups can never be all zero. This allows the wired-OR to show non-zero bits during the initial binary search.

Once the system has identified all the board IDs present, it will reactivate each board, one at a time (via the same I/O ports), and find out what resources the given board needs, what interrupt choices are available, etc. A scan is made over all the boards to collect this information.

This info is then combined with info from any ECU files on the hard disk or wired into the MLB BIOS. The ECU and BIOS PnP support for hardware on the MLB is usually synthetic, and the peripherals do not really do genuine PnP. However by examining the BIOS info plus the ECU info, the probe routines can cause the devices that are PnP to avoid those devices the probe code cannot relocate.

Then the PnP devices are visited once more and given their I/O, DMA, IRQ and Memory-map address assignments. The devices will then appear at those locations and remain there until the next reboot, although there is nothing that says you cannot move them around whenever you want.

There is a lot of oversimplification above, but you should get the general idea.

Microsoft took over some of the primary printer status ports to do PnP, on the logic that no boards decoded those addresses for the opposing I/O cycles. I found a genuine IBM printer board that did decode writes of the status port during the early PnP proposal review period, but MS said "tough". So they do a write to the printer status port for setting addresses, plus that use that address + `0x800`, and a third I/O port for reading that can be located anywhere between `0x200` and `0x3ff`.

18.12. Can you assign a major number for a device driver I have written?

FreeBSD-CURRENT after February 2003 has a facility for dynamically and automatically allocating major numbers for device drivers at runtime. This mechanism is highly preferred to the older procedure of statically allocating device numbers. Some comments on this subject can be found in src/sys/conf/majors.

If you are forced for some reason to use a static major number, the procedure for obtaining one depends on whether or not you plan on making the driver publicly available. If you do, then please send us a copy of the driver source code, plus the appropriate modifications to files.i386, a sample configuration file entry, and the appropriate MAKEDEV(8) code to create any special files your device uses. If you do not, or are unable to because of licensing restrictions, then character major number 32 and block major number 8 have been reserved specifically for this purpose; please use them. In any case, we would appreciate hearing about your driver on the ηλεκτρονική λίστα τεχνικών συζητήσεων του FreeBSD.

18.13. What about alternative layout policies for directories?

In answer to the question of alternative layout policies for directories, the scheme that is currently in use is unchanged from what I wrote in 1983. I wrote that policy for the original fast filesystem, and never revisited it. It works well at keeping cylinder groups from filling up. As several of you have noted, it works poorly for find. Most filesystems are created from archives that were created by a depth first search (aka ftw). These directories end up being striped across the cylinder groups thus creating a worst possible scenario for future depth first searches. If one knew the total number of directories to be created, the solution would be to create (total / fs_ncg) per cylinder group before moving on. Obviously, one would have to create some heuristic to guess at this number. Even using a small fixed number like say 10 would make an order of magnitude improvement. To differentiate restores from normal operation (when the current algorithm is probably more sensible), you could use the clustering of up to 10 if they were all done within a ten second window. Anyway, my conclusion is that this is an area ripe for experimentation.

Kirk McKusick, September 1998

18.14. How can I make the most of the data I see when my kernel panics?

[This section was extracted from a mail written by Bill Paul <wpaul@FreeBSD.org> on the freebsd-current mailing list by Dag-Erling Smørgrav <des@FreeBSD.org>, who fixed a few typos and added the bracketed comments]

```
From: Bill Paul wpaul@skynet.ctr.columbia.edu
Subject: Re: the fs fun never stops
To: Ben Rosengart
Date: Sun, 20 Sep 1998 15:22:50 -0400 (EDT)
Cc: current@FreeBSD.org
```

Ben Rosengart posted the following panic message]

```
 Fatal trap 12: page fault while in kernel mode
 fault virtual address   = 0x40
 fault code              = supervisor read, page not present
 instruction pointer     = 0x8:0xf014a7e5
                                ^^^^^^^^^^
 stack pointer           = 0x10:0xf4ed6f24
 frame pointer           = 0x10:0xf4ed6f28
 code segment            = base 0x0, limit 0xfffff, type 0x1b
                         = DPL 0, pres 1, def32 1, gran 1
 processor eflags        = interrupt enabled, resume, IOPL = 0
 current process         = 80 (mount)
 interrupt mask          =
 trap number             = 12
 panic: page fault
```

[When] you see a message like this, it is not enough to just reproduce it and send it in. The instruction pointer value that I highlighted up there is important; unfortunately, it is also configuration dependent. In other words, the value varies depending on the exact kernel image that you are using. If you are using a GENERIC kernel image from one of the snapshots, then it is possible for somebody else to track down the offending function, but if you are running a custom kernel then only *you* can tell us where the fault occurred.

What you should do is this:

1. Write down the instruction pointer value. Note that the `0x8:` part at the beginning is not significant in this case: it is the `0xf0xxxxxx` part that we want.
2. When the system reboots, do the following:

   ```
   % nm -n /kernel.that.caused.the.panic | grep f0xxxxxx
   ```

where `f0xxxxxx` is the instruction pointer value. The odds are you will not get an exact match since the symbols in the kernel symbol table are for the entry points of functions and the instruction pointer address will be somewhere inside a function, not at the start. If you do not get an exact match, omit the last digit from the instruction pointer value and try again, i.e.:

```
% nm -n /kernel.that.caused.the.panic | grep f0xxxxx
```

If that does not yield any results, chop off another digit. Repeat until you get some sort of output. The result will be a possible list of functions which caused the panic. This is a less than exact mechanism for tracking down the point of failure, but it is better than nothing.

I see people constantly show panic messages like this but rarely do I see someone take the time to match up the instruction pointer with a function in the kernel symbol table.

The best way to track down the cause of a panic is by capturing a crash dump, then using gdb(1) to generate a stack trace on the crash dump.

In any case, the method I normally use is this:

1. Set up a kernel config file, optionally adding `options DDB` if you think you need the kernel debugger for something. (I use this mainly for setting breakpoints if I suspect an infinite loop condition of some kind.)
2. Use `config -g KERNELCONFIG` to set up the build directory.
3. `cd /sys/compile/KERNELCONFIG; make`
4. Wait for kernel to finish compiling.
5. `make install`
6. reboot

The make(1) process will have built two kernels. kernel and kernel.debug. kernel was installed as /kernel, while kernel.debug can be used as the source of debugging symbols for gdb(1).

To make sure you capture a crash dump, you need edit /etc/rc.conf and set `dumpdev` to point to your swap partition. This will cause the rc(8) scripts to use the dumpon(8) command to enable crash dumps. You can also run dumpon(8) manually. After a panic, the crash dump can be recovered using savecore(8); if `dumpdev` is set in /etc/rc.conf, the rc(8) scripts will run savecore(8) automatically and put the crash dump in /var/crash.

FreeBSD crash dumps are usually the same size as the physical RAM size of your machine. That is, if you have 64MB of RAM, you will get a 64MB crash dump. Therefore you must make sure there is enough space in /var/crash to hold the dump. Alternatively, you run savecore(8) manually and have it recover the crash dump to another directory where you have more room. It is possible to limit the size of the crash dump by using `options MAXMEM=(foo)` to set the amount of memory

> the kernel will use to something a little more sensible. For example, if you have 128MB of RAM, you can limit the kernel's memory usage to 16MB so that your crash dump size will be 16MB instead of 128MB.

Once you have recovered the crash dump, you can get a stack trace with gdb(1) as follows:

```
% gdb -k /sys/compile/KERNELCONFIG/kernel.debug /var/crash/vmcore.0
(gdb) where
```

Note that there may be several screens worth of information; ideally you should use script(1) to capture all of them. Using the unstripped kernel image with all the debug symbols should show the exact line of kernel source code where the panic occurred. Usually you have to read the stack trace from the bottom up in order to trace the exact sequence of events that lead to the crash. You can also use gdb(1) to print out the contents of various variables or structures in order to examine the system state at the time of the crash.

Now, if you are really insane and have a second computer, you can also configure gdb(1) to do remote debugging such that you can use gdb(1) on one system to debug the kernel on another system, including setting breakpoints, single-stepping through the kernel code, just like you can do with a normal user-mode program. I have not played with this yet as I do not often have the chance to set up two machines side by side for debugging purposes.

[Bill adds: "I forgot to mention one thing: if you have DDB enabled and the kernel drops into the debugger, you can force a panic (and a crash dump) just by typing 'panic' at the ddb prompt. It may stop in the debugger again during the panic phase. If it does, type 'continue' and it will finish the crash dump." -ed]

18.15. Why has dlsym() stopped working for ELF executables?

The ELF toolchain does not, by default, make the symbols defined in an executable visible to the dynamic linker. Consequently `dlsym()` searches on handles obtained from calls to `dlopen(NULL, flags)` will fail to find such symbols.

If you want to search, using `dlsym()`, for symbols present in the main executable of a process, you need to link the executable using the `-export-dynamic` option to the ELF linker (ld(1)).

18.16. How can I increase or reduce the kernel address space?

By default, the kernel address space is 256 MB on FreeBSD 3.X and 1 GB on FreeBSD 4.X. If you run a network-intensive server (e.g. a large FTP or HTTP server), you might find that 256 MB is not enough.

So how do you increase the address space? There are two aspects to this. First, you need to tell the kernel to reserve a larger portion of the address space for itself. Second, since the kernel is loaded

at the top of the address space, you need to lower the load address so it does not bump its head against the ceiling.

The first goal is achieved by increasing the value of `NKPDE` in src/sys/i386/include/pmap.h. Here is what it looks like for a 1 GB address space:

```
#ifndef NKPDE
#ifdef SMP
#define NKPDE                   254     /* addressable number of page tables/pde's */
#else
#define NKPDE                   255     /* addressable number of page tables/pde's */
#endif  /* SMP */
#endif
```

To find the correct value of `NKPDE`, divide the desired address space size (in megabytes) by four, then subtract one for UP and two for SMP.

To achieve the second goal, you need to compute the correct load address: simply subtract the address space size (in bytes) from 0x100100000; the result is 0xc0100000 for a 1 GB address space. Set LOAD_ADDRESS in src/sys/i386/conf/Makefile.i386 to that value; then set the location counter in the beginning of the section listing in src/sys/i386/conf/kernel.script to the same value, as follows:

```
OUTPUT_FORMAT("elf32-i386", "elf32-i386", "elf32-i386")
OUTPUT_ARCH(i386)
ENTRY(btext)
SEARCH_DIR(/usr/lib); SEARCH_DIR(/usr/obj/elf/home/src/tmp/usr/i386-unknown-
freebsdelf/lib);
SECTIONS
{
  /* Read-only sections, merged into text segment: */
  . = 0xc0100000 + SIZEOF_HEADERS;
  .interp     : { *(.interp)    }
```

Then reconfig and rebuild your kernel. You will probably have problems with ps(1) top(1) and the like; `make world` should take care of it (or a manual rebuild of libkvm, ps(1) and top(1) after copying the patched pmap.h to /usr/include/vm/.

the size of the kernel address space must be a multiple of four megabytes.

[David Greenman <dg@FreeBSD.org> adds: *I think the kernel address space needs to be a power of two, but I am not certain about that. The old(er) boot code used to monkey with the high order address bits and I think expected at least 256MB granularity.]*

Chapter 19. Ευχαριστίες

This innocent little Frequently Asked Questions document has been written, rewritten, edited, folded, spindled, mutilated, eviscerated, contemplated, discombobulated, cogitated, regurgitated, rebuilt, castigated, and reinvigorated over the last decade, by a cast of hundreds if not thousands. Repeatedly.

We wish to thank every one of the people responsible, and we encourage you to to join them in making this FAQ even better.

Bibliography

[biblio-unleashed] FreeBSD Unleashed. Michael Urban und Brian Tiemann. Sams. Erste Ausgabe. 992 Seiten. Oktober 2001. ISBN 0-67232-206-4.

[biblio-44sysman] 4.4BSD System Manager's Manual. Computer Systems Research Group, University of California, Berkeley. O'Reilly and Associates. Erste Ausgabe. Juni 1994. 804 Seiten. ISBN 1-56592-080-5.

[biblio-44userman] 4.4BSD User's Reference Manual. Computer Systems Research Group, University of California, Berkeley. O'Reilly and Associates. Erste Ausgabe. Juni 1994. 905 Seiten. ISBN 1-56592-075-9.

[biblio-44suppman] 4.4BSD User's Supplementary Documents. Computer Systems Research Group, University of California, Berkeley. O'Reilly and Associates. Erste Ausgabe. Juni 1994. 712 Seiten. ISBN 1-56592-076-7.

[biblio-44progman] 4.4BSD Programmer's Reference Manual. Computer Systems Research Group, University of California, Berkeley. O'Reilly and Associates. Erste Ausgabe. Juni 1994. 866 Seiten. ISBN 1-56592-078-3.

[biblio-44progsupp] 4.4BSD Programmer's Supplementary Documents. Computer Systems Research Group, University of California, Berkeley. O'Reilly and Associates. Erste Ausgabe. Juni 1994. 596 Seiten. ISBN 1-56592-079-1.

[biblio-44kernel] The Design and Implementation of the 4.4BSD Operating System. M. K. McKusick, Kirk Marshall, Keith Bostic, Michael J Karels und John Quarterman. Addison-Wesley. Reading MA . 1996. ISBN 0-201-54979-4.

[biblio-freebsdkernel] The Design and Implementation of the FreeBSD Operating System. M. K. McKusick und George V. Neville-Neil. Addison-Wesley. Boston MA . 2004. ISBN 0-201-70245-2.

[biblio-nemeth3rd] Unix System Administration Handbook. Evi Nemeth, Garth Snyder, Scott Seebass, Trent R. Hein und John Quarterman. Prentice-Hall. Dritte Ausgabe. 2000. ISBN 0-13-020601-6.

[lehey3rd] The Complete FreeBSD. Greg Lehey. Walnut Creek. Dritte Ausgabe. Juni 1999. 773 Seiten. ISBN 1-57176-246-9.

[McKusick et al, 1994] Berkeley Software Architecture Manual, 4.4BSD Edition. M. K. McKusick, M. J. Karels, S. J. Leffler, W. N. Joy und R. S. Faber. 5:1-42.

[biblio-ja-fbsdpc98] FreeBSD for PC 98'ers (in Japanisch). SHUWA System Co, LTD.. ISBN 4-87966-468-5 C3055 P2900E.

[biblio-ja-fbsd] FreeBSD (in Japanisch). CUTT. ISBN 4-906391-22-2.

[biblio-ja-compintro] Complete Introduction to FreeBSD (in Japanisch). Shoeisha Co., Ltd. ISBN 4-88135-473-6 P3600E.

[biblio-ja-unixstarterkit] Personal UNIX Starter Kit FreeBSD (in Japanisch). ASCII. ISBN 4-7561-1733-3 P3000E.

[biblio-ja-fbsdhb] FreeBSD Handbook (Japanische Übersetzung). ASCII. ISBN 4-7561-1580-2 P3800E.

[biblio-ge-fbsdmitmeth] FreeBSD mit Methode (in Deutsch). Computer und Literature Verlag/Vertrieb Hanser. 1998. ISBN 3-932311-31-0.

[biblio-ja-fbsdinstandutil] FreeBSD install and Utilization Manual (in Japanisch). Mainichi Communications Inc..

[biblio-indo-intserv] Building Internet Server with FreeBSD (in Indonesisch). Elex Media Komputindo. Onno W Purbo, Dodi Maryanto, Syahrial Hubbany und Widjil Widodo.

[biblio-fbsdcorpnetguide] The FreeBSD Corporate Networker's Guide. Addison-Wesley.

[biblio-unixnutshell] UNIX in a Nutshell. O'Reilly & Associates, Inc.. 1990. ISBN 093717520X.

[biblio-cantfindadmin] What You Need To Know When You Can't Find Your Unix System Administrator. O'Reilly & Associates, Inc.. 1995. Linda Mui. ISBN 1-56592-104-6.

[biblio-ja-fbsdusrrefman] FreeBSD User's Reference Manual (Japanische Übersetzung). Mainichi Communications Inc.. Jpman Project, Japan FreeBSD Users Group. 1998. ISBN 4-8399-0088-4 P3800E.

[biblio-newcomeunix] Online Guide for newcomers to the UNIX environment". Edinburgh University.

[biblio-dnsandbind] DNS and BIND. O'Reilly & Associates, Inc. ISBN 1-56592-512-2. Paul Albitz Albitz und Cricket Liu. 1998. Dritte Ausgabe.

[biblio-sendmail] Sendmail. O'Reilly & Associates, Inc. 1997. Zweite Auflage. Brian Costales. ISBN 1-56592-222-0.

[biblio-esssysadmin] Essential System Administration. Æleen Frisch. Zweite Auflage. O'Reilly & Associates. 1995. ISBN 1-56592-127-5.

[biblio-tcpipnetworkadministration] TCP/IP Network Administration. Craig Hunt. Zweite Auflage. O'Reilly & Associates, Inc. 1997. ISBN 1-56592-322-7.

[biblio-managingnfsandnis] Managing NFS and NIS. Hal Stern. O'Reilly & Associates, Inc. 1991. ISBN 0-937175-75-7.

[biblio-jpmanprojectjfug] FreeBSD System Administration's Manual. Jpman Project, Japan FreeBSD Users Group. Mainichi Communications Inc.. 1998. ISBN 4-8399-0109-0 P3300E.

[biblio-xwinsystoolkit] X Window System Toolkit. Digital Press. Paul Asente. ISBN 1-55558-051-3.

[biblio-carefman] C: A Reference Manual. Prentice Hall. 1995. Vierte Auflage. Samuel P. Harbison und Guy L. Jr. Steele. ISBN 0-13-326224-3.

[biblio-thecproglang] The C Programming Language. Prentice Hall. 1998. Brian Kernighan und Dennis Ritchie. ISBN 0-13-110362-9.

[biblio-portingunixsoft] Porting UNIX Software. Greg Lehey. O'Reilly & Associates, Inc.. 1995. ISBN 1-56592-126-7.

[biblio-thestandardclibrary] The Standard C Library. Prentice Hall. 1992. P. J. Plauger. ISBN 0-13-131509-9.

[biblio-advprogintheunixenv] Advanced Programming in the UNIX Environment. Addison-Wesley. 1992. W. Richard Stevens. ISBN 0-201-56317-7.

[biblio-unixnetprog] UNIX Network Programming. W. Richard Stevens. Prentice Hall. 1998. Zweite Auflage. ISBN 0-13-490012-X.

[biblio-writeserialdriverforunix] Writing Serial Drivers for UNIX. Bill Wells. Dezember 1994. Dr. Dobb's Journal. pp68-71, pp97-99.

[biblio-unixsysarch] UNIX System Architecture. Prentice-Hall, Inc. 1990. Prabhat K. Andleigh. ISBN 0-13-949843-5.

[biblio-portingunixtothe386] Porting UNIX to the 386. William Jolitz. Dr. Dobb's Journal. Januar 1991 - Juli 1992.

[biblio-tcpipillv1theprotocols] TCP/IP Illustrated, Volume 1: The Protocols. W. Richard Stevens. Addison-Wesley. 1996. ISBN 0-201-63346-9.

[biblio-unixsysformodrnarch] Unix Systems for Modern Architectures. Addison-Wesley. Curt Schimmel. 1994. ISBN 0-201-63338-8.

[biblio-tcpipillvol3] TCP/IP Illustrated, Volume 3: TCP for Transactions, HTTP, NNTP and the UNIX Domain Protocols. Addison-Wesley. 1996. W. Richard Stevens. ISBN 0-201-63495-3.

[biblio-unixinternthenewfrontiers] UNIX Internals — The New Frontiers. Uresh Vahalia. Prentice Hall. 1996. ISBN 0-13-101908-2.

[biblio-tcpipillvol2theimplementation] TCP/IP Illustrated, Volume 2: The Implementation. Gary R. Wright und W. Richard Stevens. 1995. Addison-Wesley. ISBN 0-201-63354-X.

[biblio-firewallsandinternetsecurity] Firewalls and Internet Security: Repelling the Wily Hacker. William R. CHeswick und Steven M. Bellovin. Addison-Wesley. 1995. ISBN 0-201-63357-4.

[biblio-practicalunixsecurity] Practical UNIX Security. Simson Garfinkel und Gene Spafford. 1996. Zweite Auflage. O'Reilly & Associates, Inc. ISBN 1-56592-148-8.

[biblio-pgpprettygoodprivacy] PGP Pretty Good Privacy. Simson Garfinkel. O'Reilly & Associates, Inc. 1995. ISBN 1-56592-098-8.

[biblio-pentiumprocarch] Pentium Processor System Architecture. Don Anderson und Tom Shanley. Addison-Wesley. 1995. Zweite Auflage. ISBN 0-201-40992-5.

[biblio-progguidetothesvgacards] Programmer's Guide to the EGA, VGA, and Super VGA Cards. Richard F. Ferraro. Dritte Ausgabe. Addison-Wesley. 1995. ISBN 0-201-62490-7.

[biblio-80486] 80486 System Architecture. Tom Shanley. Addison-Wesley. 1995. Dritte Ausgabe. ISBN

0-201-40994-1.

[biblio-isasysarch] ISA System Architecture. Tom Shanley. Addison-Wesley. Dritte Ausgabe. 1995. ISBN 0-201-40996-8.

[biblio-pcisysarch] PCI System Architecture. Tom Shanley. Addison-Wesley. 1995. Dritte Ausgabe. ISBN 0-201-40993-3.

[biblio-theundocumentedpc] The Undocumented PC. Frank Van Gilluwe. Addison-Wesley. 1994. ISBN 0-201-62277-7.

[biblio-bellsystemtechnicaljournal] Bell System Technical Journal, Unix Time-Sharing System. American Telephone & Telegraph Company. Juli - August 1978. Vol 57, No 6, Part 2. ISSN0005-8580.

[biblio-commentaryonunix] Lion's Commentary on UNIX. John Lion. ITP Media Group. 1996. Sechste Ausgabe. ISBN 1573980137.

[biblio-newhackerdict] The New Hacker's Dictionary. Eric S. Raymond. MIT Press. 1996. Dritte Ausgabe. ISBN 0-262-68092-0.

[biblio-aqtrcentofunix] A quarter century of UNIX. Peter H. Salus. Addison-Wesley. 1994. ISBN 0-201-54777-5.

[biblio-unixhatershandbook] The UNIX-HATERS Handbook. Steven Strassman, Daniel Weise und Simon Garfinkel. IDG Books Worldwide, Inc. 1994. ISBN 1-56884-203-1.

[biblio-lifewithunix] Life with UNIX - special edition. Don Libes und Sandy Ressler. Prentice-Hall. 1989. ISBN 0-13-536657-7.

[biblio-bsdfamilytree] The BSD Family Tree. 1997.

[absolutebsd] Absolute BSD. Michael Lucas. No Starch Press. Juni 2002. ISBN 1-886411-74-3.

[biblio-ccppusersjournal] The C/C++ Users Journal. R&D Publications Inc.. ISSN 1075-2838.

[biblio-sysadminthejournalforunixsysadmins] Sys Admin - The Journal for UNIX System Administrators. Miller Freeman, Inc. ISSN 1061-2688.