

מסמך זה נכתב על ידי newbie בשביל newbies מתוך עניין, אמונה ואהבה ללינוקס וכמובן עקב מחסור רב באינפורמציה כתובה עיברית ברורה פשוטה ועיניינית לגבי מערכת ההפעלה לינוקס.

את המקור של תרגום זה ניתן להשיג ב:

<http://sunsite.dk/linux-newbie>

התרגום נעשה בהסכמתם של המחברים המקוריים:

Copyright (c) <1999-2002> by Peter and Stan Klimas.

ותחת ה- Open Publication License v1.0, 8 עם השינויים שמצויינים ב Inag_licence

(http://sunsite.dk/linux-newbie/Inag_licence.html)

הערות תיקונים שאלות ביקורת וכו' ייתקבלו בברכה ב etay218@bezeqint.net
כל עוד זה תלוי במתרגם, קוראי מסמך זה יכולים לעשות בו ככל העולה על רוחם.

אלעד איתי

תודה מיוחדת לאמיר טל מ www.whatsup.org.il

ולרן צחור-מנהל פורום לינוקס בynet- שאם לא היה עונה לי על שאלה ששאלתי, ספק אם הייתי כותב שורות אלו.

"software is like sex its better when its free"

חלק 0. לאלה שעדיין לא החליטו

תוכן החלק הזה:

- 0.1 ועכשיו ברצינות, למה לינוקס? [עמ' 2]
- 0.2 האם לינוקס בשבילי? [עמ' 2]
- 0.3 לינוקס קשה למתחילים [עמ' 2]
- 0.4 מה הוא הערך המוסף של לינוקס? [עמ' 3]
- 0.5 מה הם ההבדלים בין לינוקס ליוניקס? [עמ' 5]
- 0.6 מה הם ההבדלים בין לינוקס ומייקרוסופט ווינדוס? [עמ' 5]
- 0.7 אני לא מאמין בתוכנה חינם וכו' [עמ' 6]
- 0.8 אין דבר כזה ארוחות חינם [עמ' 6]
- 0.9 אני זקוק לאבטחה ברמה גבוהה, עם תוכנות מסחריות, אני יכול לתבוע אותם אם משהו מתפקשש [עמ' 7]
- 0.10 אני צריך סטנדרטים. תאגידי תוכנה גדולים (מייקרוסופט) מספקים סטנדרטים [עמ' 7]
- 0.11 הפופולריות של מייקרוסופט ווינדוס מבטיחה שהיא "כאן כדי להישאר" [עמ' 9]
- 0.12 אבל לינוקס מסתעפת לכמה מערכות שונות [עמ' 9]
- 0.13 לינוקס זה קאלט [עמ' 9]
- 0.14 העלות הכוללת הסופית של הבעלות על לינוקס גבוהה [עמ' 10]
- 0.15 לינוקס חולמת בהקיץ; עסקים זה מה ששולט היום בעולם [עמ' 11]
- 0.16 לינוקס זה חרטא וכו' [עמ' 11]

0.1 ועכשיו ברצינות, למה לינוקס?

אם אתה באמת נהנה לעבוד עם מחשבים, לינוקס היא מערכת ההפעלה של חלומותיך. היא מהנה הרבה יותר מכל מערכת הפעלה אחרת בסביבה. בכל מקרה, הסיבה שבגללה לינוקס באמת מהפכנית היא בגלל שהיא תוכנה פתוחה. המדע והטכנולוגיה שלנו עובדים הודות לזמינות החופשית של המידע והאפשרות לצפות האחד בעבודתו של השני. האם היית טס במטוס שנבנה על בסיס סוד מדעי ומידע מוסתר של איש פרט למתכנן המטוס אסור להסתכל בקרבו? ובכן מדוע שתבטח בתוכנת מחשב שמכילה חלקים ואלגוריתמים סודיים? מערכת ההפעלה לינוקס בעלת הקוד הפתוח אידיאלית לאפליקציות חשובות וקריטיות-- יכולות האבטחה והעוצמה שלה מבוססות על פתרונות חזקים שכל אחד יכול להסתכל בהם, לבקר או לשפר אותם. זוהי ההטמעה של הגישה המדעית אל תוך המיחשוב.

ייצור פרסות סוסים, זכוכית טובה, או מדידת זמן היו פעם סודות מסחריים שמורים היטב. המדע והטכנולוגיה התפוצצו לפני 500 שנה הודות לשיתוף הידע על ידי הדפוס בימים הראשונים של הדפוס רבים מאלו שהעזו לשותף נרדפו בגין חשיפת סודות מסחריים. לינוקס לעידן המחשב היא מה שגוטנברג היה לכתובה. בתקווה שאיש לא יירדף הפעם. לינוקס אכן מתנגשת עם אלו שטוענים לבעלות של מידע, שמנסים להחזיר את הזמן 500 שנים אחורה.

0.2 האם לינוקס בשבילי?

רק אתה יכול לענות על השאלה הזו. לינוקס היא מערכת הפעלה דמויית יוניקס, בוגרת, עוצמתית מאובטחת ובעל יכולות רבות. העוצמה וריבוי היכולות שלה באות יחד עם מחיר-- ייתכן ותצטרך לדעת קצת קורא וכתוב במחשבים כדי להקים את לינוקס ולתחזק אותה. באופן יחסי לינוקס קלה לשימוש באופן כזה שברגע שהתקנת את האפליקציות ואת מערכת ההפעלה כמו שצריך גם אמא שלך תוכל להשתמש בה, במידה והכנת בשבילה חשבון גרפי והנחת את האייקונים/תפריטים המתאימים על גבי שולחן העבודה שלה. לינוקס מאובטחת, ככה שאמא שלך לא תוכל לפגוע במערכת לא משנה מה היא תנסה לעשות, אלא אם כן היא תשתמש בפטיש.

לינוקס די שונה ממייקרוסופט ווינדוס, אז אל תצפה שהיא תהיה ברורה וקלה לך אם אתה מכיר קצת ווינדוס. ייתכן ותצטרך ללמוד קצת. מצד שני אם אתה מגיע מיוניקס, לינוקס תהיה קלה בשבילך אם אינך יודע הרבה על מחשבים או שאתה לא נהנה מהם, רוב הסיכויים שאדמינסטריית לינוקס אינה בשבילך.

אם אתה לא מכיר את החומרה שלך התקנת לינוקס יכולה להיות אתגר.

0.3 לינוקס קשה למתחילים

יש בזה משהו. אבל השאלה האמיתית היא: האם אתה רוצה ללמוד אותה? איש מהמחברים (או מהמתרגמים) אין רקע במדעי המחשב, ועדיין אנחנו משתמשים בלינוקס כל יום ואנחנו אוהבים את זה.

0.4? מהו הערך המוסף של לינוקס

לינוקס יכולה לתת לך:

*סביבת משתמש מודרנית, יציבה מאוד, בעלת יכולת של ריבוי משימות על גבי חומרת מחשב לא יקרה, ללא (או כמעט ללא) תשלום תמורת התוכנה. לינוקס היא פלטפורמה עשירה ועוצמתית--אל תחשוב שהיא מערכת הפעלה לעניים. ללינוקס מהקופסה יש יכולות כמו למייקרוסופט ווינדוס NT עם \$5000 תוספות תוכנה. היא מאוד יציבה, ודורשת חומרה פחות חזקה מאשר ווינדוס למשימות זהות.

*עוצמת עיבוד אולטימטיבית, ניידות, וגמישות. אשכול מחשבים של לינוקס הביס (אפריל 1999) מחשב על בבנצ' מרק סטנדרטי.

*לינוקס מאוד "מסטונדרטת" --היא POSIX, תואמת UNIX. כן, לינוקס היא צאצא משובח של יוניקס. המילה יוניקס לא נאמרת יחד עם לינוקס מכיוון ש"UNIX", זה סימן רשום.

לינוקס פופולרית מאוד על גבי מחשבים מבוססי אינטל (מחיר), אבל רצה היטב גם על מספר רב של פלטפורמות חומרה, ממחשבים דמויי צעצוע עד למחשבי מיינפריים. הפצה אחת (דביין-DEBIAN) מבטאת את הרעיון בצורה הבאה:

"לינוקס, מערכת ההפעלה האוניברסלית"

*פלטפורמת לימוד גדולה באמת. אם אתה הורה, אתה אמור לשמוח אם בנך/בתך משתמשים בלינוקס. היא/הוא באמת יילמדו משהו בעל ערך מוסף שילווה אותם לאורך זמן. אם אתה מורה, כדאי שתשקול התקנת לינוקס בבית הספר שלך. "זהו אכן עולם מוזר בו מחנכים צריכים להשתכנע לגבי העובדה ששיתוף מידע, בניגוד לחסימת גישה אל מידע הוא אכן דבר חיובי"

<http://edge-op.org/grouch/schools.html>

אתה בוחר בלינוקס אם אתה מעוניין לספק השכלה ולא אימון או הכשרה. כמה שהאוניברסיטה טובה יותר, ככה הסיכוי שמחלקת המחשבים שלה משתמשת בלינוקס ללימוד. תחת לינוקס אתה יכול למשל, להתחיל לשנות לערוך ולקמפל בעצמך אפליקציית גיליון נתונים מורכבת בדיוק כמו מייקרוסופט אקסל. לינוקס מניחה אותך ישר בחזית בטכנולוגיה, בניהול פרויקטים, ומתודולוגיית מדעים (מורים רבים לא ישתמשו בלינוקס בבית הספר מכיוון שההשכלה שלהם בתחום המחשבים לוקה בחסר. זה לפחות מה שאני רואה)

*יכולות רשת מעולות בנויות אל תוך מערכת ההפעלה שלך. אתה חושב שאין לך צורך ברשת?

ברגע שתתנסה ברשת ביתית לא תוכל לחיות בלעדיה. תוכל לשתף כוננים בין מחשבים כרטיסי קול מודמים, מדפסות וכו'. תוכל לגלוש באינטרנט דרך שני מחשבים באותו הזמן תוך שימוש בחיבור יחיד לאינטרנט. תוכל לשחק משחקים עם הבן שלך. ואפילו ה-386 הישן שלך עם ווינדוס 3.11 יקום לתחיה, כשתחבר אותו לשרת לינוקס פנטיום שלך. כל תוכנות הרשת הדרושות מגיעות עם לינוקס כסטנדרט בחינם, רק התקנה והקמה דרושות, וזו אינה תוכנה סוג ב'--זו בדיוק אותה תוכנה שמריצה את רוב האינטרנט(שרת הווב אפאצ'י - APACHE מריץ יותר מ-50% מכל שרתי הווב באינטרנט וסנדמייל SENDMAIL נוגע בכ-70% מכל הדואר האלקטרוני). את ההנאה שברשת ביתית יכולתי לגלות רק על ידי שימוש בלינוקס.

*קישורים לרשתות מייקרוסופט, נובל, ואפל. קריאה/כתיבה אל תוך פורמטי דיסק של מייקרוסופט ופורמטים אחרים. זה כולל שימוש שקוף במידע שמאוחסן על מחיצות ווינדוס בכונן שלך.

*מאות אפליקציות, חופשיות, במגוון תחומים בחינם.

*מאות רבות של אפליקציות מיוחדות/מתמחות שנבנו על ידי חוקרים ברחבי העולם (אסטרונומיה, טכנולוגיית מידע, כימיה, פיסיקה, הנדסה, בלשנות, ביולוגיה...) בתחומים רבים לינוקס נראית כמערכת ההפעלה היחידה שקיימת (נסה לברר מה חברך האסטרונום מריץ על המחשב שלו). התוכנה בקטגורייה זו אינה כל כך קלה לשימוש, אבל אם אתה מעוניין בעוצמה, זוהי התוכנה הטובה ביותר שיש לאנושות בתחומים אלו. יש לך ספקות? תן מבט ב: <http://SAL.KachinaTech.COM/Z/2/index.html> אם אתה מעוניין בדוגמאות.

*אלפי כלים, יישומים, ותוכניות קטנות. "קטן זה יפה" הולך טוב עם הפלוסופיה של לינוקס. הכלים והתוכניות הקטנות של לינוקס עובדות לעיתים אחת מאחורי השניה כדי לבצע משימות מורכבות.

*גירסאות של כל התוכניות המסחריות הטבות ביותר כולל כל מסדי הנתונים הגדולים (אורקל סיבייס אבל לא מייקרוסופט). רבות מתוכניות אלו (כולן?) מוצעות בחינם למפתחים ושימוש אישי.

*פלטפורמת פיתוח מלאכת מחשבת עם שפות תיכנות הטובות מסוגן וכלים שמגיעים חופשי עם מערכת ההפעלה. גישה אל קוד המקור של מערכת ההפעלה, במידה ואתה זקוק לו, גם הוא מגיע חינם. הקומפיילר של שפת C שמגיעה כסטנדרט עם לינוקס יכול לקמפל קוד לרוב הפלטפורמות כנראה, יותר מכל קומפיילר אחר עליי האדמות. PERL, PYTHON, GUILLE, TCL, RUBY scripting "SHELL" עוצמתי ואפילו אסמבלר מגיע כסטנדרט עם לינוקס.

*חופש יחסי מוירוסים, סוסים טרויינים, "פיצ'רים" של יצרני תוכנה, חדירה אל הפרטיות, שידרוגים כפויים, פורמטי קבצים סגורים, סכמות רישוי ושיווק, רישום מוצר, מחירים גבוהים, ופירטיות. איך זה נשמע לך?

*פלטפורמת ההפעלה שמובטח שביא "כאן כדי להישאר". מכיון שלינוקס אינה בבעלות של איש, היא אינה יכולה לצאת אל מחוץ לעסקים. הרשיון הציבורי הכללי ה-GPL= GENERAL PUBLIC LICENSE-מבטיח שפיתוח/תחזוקה ייסופקו כל עוד ישנם משתמשי לינוקס. ישנו מספר גדול של משתמשי לינוקס משכילים מאוד ועשרות אלפים של פרויקטים שנמצאים בפיתוח.

*פלטפורמה שתפתח טכנולוגית בקצב מהיר. זה מובטח על ידי מודל פיתוח מודרני מבוסס קוד פתוח שלינוקס מיישמת: "נבנה על גב המפתח הקודם" וגם "עמית מסתכל בקוד שלך" בניגוד לגישה האנכרוניסטית הסגורה "תמיד תתחיל מכלום" ו"אף אחד לא יראה את הקוד שלי". אפילו אם האופנה האחרונה הלכה לעולמה לינוקס תמשיך להתפתח בדיוק כמו לפני שהאופנה הזו התחילה. לפיתוח בקוד פתוח יש את המוזריות שלו: הפיתוח נראה איטי למדי (במאונך) אבל הוא חל על גבי חזית רחבה מאוד, באגים חמורים באבטחה מתוקנים כמעט עם זמן הגילוי שלהם, ישן בדרך כלל כמה אלטרנטיבות לתוכנית או לפונקצייה מסויימת.

אם אתה מעוניין ללמוד ממקור ראשון על הרשיון הציבורי הכללי, בדוק את מסמכי ה-GNU האלה:

<http://www.gnu.org/copyleft/gpl.html>

<http://www.gnu.org/gnu/linux-and-gnu.html>

<http://www.gnu.org/philosophy/categories.html#thegnusystem>

סביר להניח שהרשיון תחתיו מופצת לינוקס הינו אחד החלקים היותר חשובים במסמך זה. הרשיון נועד לקדם את חופש המידע. פרויקטים נוספים של הקוד הפתוח כוללים מדע וחוק (בקושי בדיחה) השיטה של לינוקס אינה משהו חדש-- היא פשוט ההחלה של הגישה המדעית על תוכנה: אתה מקבל מידע באופן חופשי, מוסיף את הרעיונות שלך, מתפרנס, ולבסוף, משאיר את המידע חופשי. אבל, נראה שכמה תאגידיים גדולים ועורכי הדין שלהם מתאמצים מאוד כדי לשנות את זה, הם מנסים לדחוף את הזמן אחורה אל ימי הביניים, אל הזמנים בהם מידע היה בבעלות מישהו. לכן, אתה יכול לראות כמה אנשים שקשורים ללינוקס מעורבים בכל מיני מאבקים.

למילה "חופשי" יש שתי משמעויות די שונות בשפה האנגלית (והעיברית-א.א.), מה שמוביל לתפיסה לא נכונה את הטבע החופשי של לינוקס. שתי משמעויות אלו נובעות משם התואר הלטיני "liber" ומתואר הפועל "gratis" והן בדרך כלל מתקשרות אל המינוחים "free speech" ו-"free (of charge) beer". רוב התוכנה בלינוקס היא free בשני המובנים, אבל רק הראשון הכרחי ללינוקס.

0.5 מה הם ההבדלים בין לינוקס ליוניקס?

מבחינת שימוש בשורת הפקודה, כמעט כלום, למרות שזה משתנה (לטובה או לרעה). ללינוקס יש הרבה יותר כח משיכה שיווקי. מבחינת ממשק גרפי, ההבדלים אינם מז'וריים-- לינוקס, כמו רוב היוניקסים משתמשת במערכת x-windows.

ההבדלים העיקריים:

לינוקס חופשית, בעוד יוניקסים רבים עולים הרבה. כנ"ל לגבי אפליקציות--אפליקציות טובות רבות ניתנות להשגה באופן חופשי ללינוקס. אפילו אותה אפליקציה מסחרית (אם אתה מעוניין לקנות אחת) עולה הרבה יותר ליוניקס מסחרית מאשר ללינוקס.

לינוקס רצה על פלטפורמות חומרה רבות, בעוד יוניקס טיפוסית קשורה אל חומרה מסויימת בדרך כלל-וחומרה זו נוטה להיות יקרה הרבה יותר מאשר תואם פיסי טיפוס. עם לינוקס, אתה בשליטה על המחשב שלך, בעוד ברוב היוניקסים אתה בדרך כלל מוגדר כ"i-user" (כמה אדמיניסטרטורים מבטאים זאת "loser")-- לינוקס מרגישה קצת כמו דוס ווינדוס בשנות ה-80/90 המאוחרות, אבל היא הרבה יותר יציבה ועשירה. בעוד מערכת יוניקס טיפוסית מרגישה כמו מיינפריים משנות ה-60/70. ייתכן שיוניקסים רבים בוגרים יותר באזורים מסויימים (אבטחה למשל, כמה אפליקציות שקשורות בהנדסה, תמיכה טובה יותר בחומרה "כבדה"). לינוקס היא יותר בשביל ג'ו שרוצה להריץ את שרת משל עצמו או תחנת עבודה הנדסית.

0.6 מה הם ההבדלים בין לינוקס למייקרוסופט ווינדוס?

מבחינת הקלקה על העכבר, כמעט אין, מרגע שלינוקס מותקנת כראוי. ההבדלים העיקריים:

לינוקס חופשית, בעוד ווינדוס עולה כסף. אותו הדבר לגבי אפליקציות, פורמטי הקבצים של

לינוקס חופשיים, כך שאתה יכול לגשת אליהם במיגון דרכים. בווינדוס, העיקרון הנפוץ הוא לגרום לך לנעול את המידע שלך בפורמטים סודיים שניתן לגשת אליהם רק בעזרת כלים שהושכרו לך על ידי היצרן במחיר שלו. עד כמה מושחתים (או לא מתאימים לתפקיד?) צריכים להיות הפוליטיקאים שנועלים את המידע הציבורי שלנו בפורמטים כאלו! מה שנקבל עם מייקרוסופט זה חוזה שכירות לשלוש שנים על רשומות בריאות שאנחנו צריכים לשמור "למשך מאה שנים"

http://news.bbc.co.uk/hi/english/health/newsid_1694000/1694372.stm

עם לינוקס, הסבירות שתפר איזה שהוא הסכם רישוי, נמוכה. כל התוכנה, שלך היא. עם מייקרוסופט ווינדוס סביר להניח שכבר הפרת כל מיני הסכמי רישוי וניתן להכריז עלייך כעל פיראט מחשבים אם רק איזה עורך דין חכם ירצה בכך. מייקרוסופט ווינדוס מנסה להיות מערכת ההפעלה בעלת המכונה המשותף הנמוך ביותר (לטובה או לרעה) בעוד לינוקס בנויה למשתמשי מחשבים מתוחכמים יותר (לטובה או לרעה) מייקרוסופט ווינדוס מבוססת על דוס. לינוקס מבוססת על יוניקס. הממשק הגרפי של ווינדוס מבוסס על הנחיות של מייקרוסופט עצמה בעוד הממשק הגרפי של לינוקס מבוסס על סטנדרט התעשייה X-WINDOWS. לינוקס מרפה את ידי ווינדוס בכל מה שקשור לאפשרויות רשת, כפלטפורמת פיתוח, יכולות עיבוד נתונים, וכתחנת עבודה מדעית. לשולחן העבודה של ווינדוס יש מראה מבריק יותר, (בדוק את רדהאט 8.0-א.א) אפליקציות עסקיות חלקות יותר, ויותר משחקים לילדים (אלו לא משחקים טובים יותר, המשחקים של לינוקס נוטים להיות מתוחכמים יותר). לינוקס הרבה יותר עשירה באפשרויות מאשר תוכל לדמיין. נשמע פעם באינטרנט: "שני מוצרים גדולים הגיעו מהאוניברסיטה של קליפורניה: UNIX ו-LSD. ואני לא חושב שמדובר בצירוף מקרים."

0.7 אני לא מאמין בתוכנה חינם וכו'

והאם אתה מאמין באינטרנט? האינטרנט ולינוקס חולקים רעיונות משותפים רבים ויש להם שורשים משותפים. האם אתה זוכר את חוסר האמונה ששרה לגבי האינטרנט לפני כמה שנים, הוויכוח הבלתי? נגמר והבלתי מנוצח לגבי העובדה שאינטרנט חופשי לא יכול להתקיים. המציאות פשוטה. שיתוף פעולה ורצון טוב יכולים להועיל לרבים באותו זמן: המאמץ שלך אינו ההפסד שלי. האינטרנט עובד בסדר גמור ומתפתח בקצב מהיר. כך גם לינוקס.

לפניך דעה של מנהל ב-IBM:
"הסיבה שבגללה אנחנו כל כך מתרגשים לגבי לינוקס היא שאנחנו מאמינים שלינוקס יכולה לעשות לאפליקציות מה שהאינטרנט עשה לרשתות"
IBM הוציאה זה עתה (מאי 2002) ביליון דולר כדי לגרום ללינוקס לרוץ על כל פלטפורמות החומרה שלה (מיינפריים, תחנות עבודה, פיסי, ניידים).

0.8 אין דבר כזה ארוחות חינם?

ברט בזנט <bbazant@shaw.wave.ca> כתב ב-linux today: המשאבי תוכנה הינם דלים כי כולנו שומרים עליהם בבעלות ובסודיות. אבל לא לינוקס! כשאני נותן לך את התוכנה שלי, ייתכן מאוד שאני יוצר לעצמי הזדמנות, בכל מקרה אני מצליח לשמור עליה אפילו לאחר שנתתי לך אותה. זוהי ארוחת חינם שנחשפת בהיסטוריה רק על ידי ככרות לחם ודגים. (א.א.?????)

0.9 אני זקוק לאבטחה ברמה גבוהה. עם תוכנה מסחרית אני יכול לתבוע אותם אם משהו מתפקשש

אל תסתמך על תביעה. מדי יום ביומו, דברים מתפקשים על מכונות NT. קרא את הסכם הרשיון שיש לך ממייקרוסופט כדי לגלות שאין כל ערבות לכך שמהו יעבוד ואין נזקים שמוכרים בבתי משפט.

בכלל, נסיון לתבוע יהיה ביזבוז של כסף.

גם לינוקס לא מספקת כל ערבות, למרות שהיא הרבה יותר מאובטחת מכל גירסה של ווינדוס. אם אתה באמת רגיש לגבי אבטחה, אתה יכול להשתמש בכלי האבטחה החזקים שנבנים על ידי חברות שנסמכות על הזמינות של קוד המקור כדי לעצב את כלי האבטחה שלהן (קריפטקום Kriptkom מגרמניה- מספקת פיירוול חזק)

האבטחה העמומה שמיושמת במייקרוסופט ווינדוס הוכחה פעמים רבות כגישה נאיבית.

לפניך ציטוט מפי דויד קרטרופ, מהנדס מחקר, בוכס, גרמניה:

"העובדה הפשוטה שמכתיבה את השימוש בלינוקס ובמוצרים פתוחים אחרים ולא במייקרוסופט ווינדוס NT היא הימנעות ובריחה מסיכון. הסיכונים שאתה נוטל על יד שימוש ב-NT-הינם לחלוטין מחוץ לכלל שליטתך, ואין אף אחד שתוכל לתבוע במידה ומשהו לא מסתדר כשורה. העובדה שאנשים עדיין מאמינים במיתוס שווינדוס מציעה יותר אמינות מאשר לינוקס עדיין נשאר בגדר חידה בשבילי."

0.10 אני צריך סטנדרטים. תאגידים גדולים (מייקרוסופט) מספקים סטנדרטים

אולי זה מה שאנשים מצפים מתאגידים גדולים, אבל המציאות שונה. בעבר חברות גדולות אהבו להמציא אגוזים שיכלו להתפצח רק בחנויות השרות שלהן. האם אגוזים אלו הפכו להיות הסטנדרט? בקושי. הם לא הפכו כי לא הייתה מעורבת בכך טובת הציבור, והם לא יכלו להפוך לכאלה כי הם היו פטנט. למזלנו, היום יש לנו סטנדרטים פתוחים וחופשיים לאגוזים. "סטנדרט שנמצא בבעלות" הוא כזה אוקסימורון מגוחך שממש קשה להאמין שאנשים משכילים יכולים להאמין בו. (נכון לעכשיו, מנגנוני השיווק מכתיבים לנו את המושג "הדה פקטו סטנדרט" או "סטנדרט התעשייה" כדי לכסות על הכיעור של חוסר הסטנדרטים)

דוגמא מעולם המחשבים. הפורמט ה"סטנדרטי" של מייקרוסופט וורד שונה פעמים רבות במשך השנים האחרונות. וקרוב לוודאי שהוא ממשיך להשתנות מסיבה משתלמת כלכלית: ברגע שחברות אחרות מבצעות מה שניקרא "הינדוס הפוך" על פורמט הוורד הנוכחי, מייקרוסופט משנה אותו. ישנם אפילו תתי פורמטים (שמירה מהירה של מייקרוסופט אומר למישהו משהו?). מה גם שכל התחום הזה הוא סגור לגמרי. מייקרוסופט לא מפרסמת כל הנחיות. איך משתמש יכול להרוויח מזה בטווח הרחוק.

מה היא הערבות של מייקרוסופט שהפורמט של וורד 6 יהיה תקף גם ב2020?

קסביר בסורה:

"הסטנדרטים של מייקרוסופט הינם עריציים-החמקמקות של חוסר התאימות בין פורמט הקבצים של אופיס 97 עם גרסאות ישנות יותר או תת הגרסאות של סטנדרטים פתוחים כמו X.M.L עם תוספות בלעדיות שדורשות אינטרנט אקספלורר 5, האקטיב סרבר של מייקרוסופט, וכיצא באלו, הינם תזכורות פיכחות למה שהחברה הזו עושה לשוק..."

המונופול של מייקרוסופט לא מבטיח לך שגירסת האופיס הנוכחית שלך תעבוד עם כל גירסא קודמת או עתידית של אופיס. וזה למרות שמספר מתנצלי מייקרוסופט טוענים שההטבה מהמונופול של מייקרוסופט היא קביעת הסטנדרט בייצור אפליקציות.

כדי להוסיף על הביבלול, חברות בדרך כלל לא "מסטנדרות" את פורמטי הקבצים שלהם אלא רק לאפליקציות שאמורות לתמוך בהן. זה כמו לקבוע סטנדרט לייצרן אגוזים במקום על האגוזים עצמם. איך זה אמור לעבוד אם ייצור הקבצים ממשיך להשתנות במטרה מפורשת כדי להעצים את המכירות.

טד קלרק:

"אנחנו זקוקים לפורמטי קבצים פתוחים בעלי סטנדרט, כדי שמשמשים יוכלו להחליף מסמכים בין פלטפורמות. תוכנת עיבוד התמלילים ששימשה לייצרת הקבצים הללו בכלל לא אמורה להיות אישו"

ישנם כמה פורמטי קבצים נוספים מקושרי טקסט או מסמך שהינם יותר סטנדרט מפורמט הקבצים של מייקרוסופט וורד:
...ascii, xml, html, sgml, latex, tex, postscript, pdf, dvi
ולכולם יש תמיכה מצויינת תחת לינוקס. גם פורמט הקבצים של word יכול להיכתב ולהיקרא היטב תחת לינוקס.

לינוקס מטיבעה מבוססת על סטנדרטים אמיתיים, מופצים וחופשיים בגלל שהקוד הפתוח גורם לספסיפיקציות להיות זמינות לכולם (מתחרים ולא מתחרים). אני חושב שהצורך הדחוף בסטנדרט פתוח הוא אחד מהכוחות המניעים שמאחורי לינוקס. חלק מהאנשים מרגישים שהם אינם יכולים להרשות לאלגוריתמים ולנתונים שלהם להיות יישות מסחרית, ועוזבים לבד את מי שבאופן קבוע מפגין חוסר אמינות. הסתכל בטיוטה הזו של חוק ארגנטינאי כדי לטעום מן העתיד. זה נשמע כאילו הארגנטינאים יהיו הראשונים שיחליטו שהרשומות הציבוריות שלהם לא יכולות להיות מוחזקות כבנות ערובה על ידי יישות מסחרית.

"אירגונים ציבוריים לאומיים שמוזכרים במאמר 1 של חוק זה, לא יורשו להשתמש בתוכניות שמאחסנות מידע בפורמט לא ציבורי"

דעה עמוקה בקהילת המשתמשים של לינוקס טוענת שאכן יש בעייה רצינית עם סטנדרטי מיחשוב שמנוצחים על ידי יצרני תוכנה גדולים. זה כולל את הנתונים ה"סטטיים" שלנו, כמו גם את המידע המוטמע בקוד על גבי המחשב שלנו. האם אנחנו יכולים להרשות לעצמנו שיהיה מישהו אחר שיחליט בשבילנו מתי, איך, ובאיזה מחיר אנחנו יכולים לגשת לנתונים אלו? הבעייה זוכה להתעלמות ואפילו זוכה לעידוד מהאנשים שמקבלים שכר בשביל לדאוג לנושא.

אם אתה כותב תוכנת מחשב שאמורה לאחסן כל סוג של מידע לגבי החברה שלך, כדאי שתיקח את העצות הבאות בחשבון:

1. בחר בסטנדרט מבוסס קוד פתוח.

2. שמור על פשטות. בצורה זו אתה מבטיח הגנה אמיתית להשקעה שלך בתוכנה. אותו הדבר נכון לגבי בחירת הפורמטים שאתה בוחר לאיחסון המידע היקר שלך.

0.11 הפופולריות של מייקרוסופט מביטיחה שהיא כאן כדי להישאר

קרוב לוודאי שזה נכון. נניטנדו גם היא כאן כדי להישאר. בכל מקרה אני אוהב מיחשוב אז אני בוחר במערכת הפעלה עוצמתית, לא את המערכת בעלת המכנה המשותף הנמוך ביותר שמעוצבת ל"כולם".

לינוקס היא די כאן להישאר בגלל טבע הקוד הפתוח שלה (לינוקס פשוט לא יכולה להיות מחוץ לעסקים) היא הבחירה הסטנדרטית לאין ספור פרוייקטים שלא הולכים להיעלם, וחלקם אף די קריטיים. בדוק את תחנת החלל הבינלאומית, שבה לינוקס היא מערכת ההפעלה -

<http://www.linuxjournal.com/article.php?sid=3024>

ובנוסף, לעולם אל תפחית בערך כוחה של קהילת הלינוקס. לינוקס היא "כאן כדי להישאר" לפחות בשביל אוונגרד המיחשוב. לינוקסאים רבים לא אפילו לא מעוניינים שלינוקס תהיה פופולרית מאוד מכיוון שהם מפחדים שזה יכול פגוע במעמד העלית של לינוקס.

0.12 אבל לינוקס מסתעפת לכמה מערכות שונות...

סיעון נפוץ שמופץ על ידי אלו שמתמחים בשיטות שיווק הידועות כ"פחד, חוסר וודאות וספק".

מסתעפת בהקשר זה פירושה חלוקה לענפים של תוכנית מחשב. כמו גם יצירת תת גירסאות של התוכנית. ולבסוף לעשות פרגמנטציה ללינוקס.

ישנה עדות דלה (אם בכלל) לגבי סיעוף מזיק של כל תוכנה שכלולה בהפצת לינוקס אופיינית היכן שכן התרחש סיעוף, הוא תמיד הפך למועיל. די אפשרי, שזה התרחש, על אף שאין מחסומים מלאכותיים לסיעוף תוכנה תחת לינוקס, בנוסף גם אין כל מחסומים מלאכותיים שמונעים שילוב של החלקים הטובים ביותר חזרה.

הרקע התיאורתי על איך סיעוף תוכנה יכול להועיל להתפתחות שלה ניתן למעשה לפני זמן רב על ידי הפילוסוף הגרמני גורג וויליאם פרדריך הגל (1770-1831), שעם קונספט ההתפתחות הדיאלקטית שלו בתופעת הרוח אמר "כשקיים פילוג בקבוצה, נראה כי הוא מבטא את דלותה, אך דווקא את עושרה הוא מבטא"

0.13 לינוקס זה קאלט

לקהילת הלינוקס הודבקה פעמים רבות תווית של "קנאים דתיים", במיוחד על ידי עיתונאים שמגזיני המחשב המאוד מבוססים בהם הם עובדים קיבלו משוב מסיבי לגבי כתבות לא צודקות על לינוקס אכן כן, קהילת הלינוקס היא רבת מספרים, מלומדת, ומוכנה לבטא את דעותיה. הרבה מגזיני/עיתוני מחשבים יודעים שלינוקס שווה פחות כסף בשבילם (משתמשים שמשלמים פחות על כל מה שמסביב למחשבים ולציוד נלווה ודורשים יותר) האם זה מסביר את המושג קנאה דתית?

תודו בזה, אנשי המכירות שלכם מעמידים פני עיתונאים. בקושי נשארה איזו שהיא כנות בעיתונות המחשבים. כמה מילים על לינוקס פירסם הפי. סי מגזין) או מה שלא יהיה) בשנים 02-01-1999? האם לינוקס לא הייתה לפחות עניין טכנולוגי בזמן הזה? בוודאי שהיא הייתה, אבל בכל זאת זממתם להשאיר את קהל הקוראים שלכם באפלה, מוכרים את היושר

העיתונאי שלכם בעבור כסף. ועכשיו, לאחר שלינוקס צפה אל פני הזרם המרכזי בתקשורת (לא קשורת המחשבים), אתם ממשיכים לכתוב כתבות שאומרות "כן, אבל לינוקס לא תוכל / לא יכולה / אולי תוכל" ("...מנסים להשתמש בטקטיקת ה"פחד, חוסר הוודאות והספק" כדי להרוג את התופעה הזו) ובנוסף כותבים "מייקרוסופט כבר... ממשיכים לכתוב על הפלטפורמה החמקמקה, הלא יציבה היקרה וחסרת הארכיטקטורה, שההישג הגדול ביותר שלה היה רדיפת בצע שלא נשמע כמותה על ידי מניעת תאימות כוללת, והרג של כל סטנדרט מוצע (על ידי אימוצו וכינוסו לבעלותכם על יד תוספות). את מי אתם משרתים? בטוח שלא את הקוראים שלכם.

אמרתי את זה בצורה החזקה ביותר שיכולתי. האם אני קנאי? או שאני רק מנסה להביע את חוסר ההסכמה שלי לגבי מעשיכם שלא משרתים אדם מלבד את עצמכם?

אתם חושבים שסיפוק עצמי זה בסדר בעסקים? עד כמה מעורר רחמים העסק שלכם צריך להיות! אני תמיד חשבתי שעסקים זה חוזה חברתי שבו אנחנו מחליפים סחורות בעלות ערך, ככה שהתועלת תהיה הדדית. לאורך ההיסטוריה, חברות נהגו לתלות, להגלות או לערוף את ראשי אלו שבאמת התעסקו בעסקים האנוכיים שלהם בלבד. ובכן הזמנים השתנו. לטובה או לרעה, קצת לרעה)))

0.14 העלות הכוללת הסופית של הבעלות על לינוקס גבוהה

אף אחד לא יודע באמת איך לחשב" עלות כללית סופית" של חתיכת חומרה. לכן עורך דין טוב+ רואה חשבון טוב יכולים להוכיח כל דבר שהם מקבלים כסף תמורתו, והם בדרך כלל עושים את זה. תנו לי לבחון הערכה קטנה לגבי העלות הכללית הסופית שנובעת מבעלות על מייקרוסופט ווינדוס בוא ונחבר את העושר המצטבר של כל יוצרי ווינדוס, נוסף את כל המשכורות של כל מתכנתי ווינדוס, היועצים, אנשי תמיכה והוראה, ניהול וכו' ועכשיו נוסף את ההפסדים שסביר להניח שלקוחות נחלו בזמן שתאגידי תוכנה הציגו להם פיצ'רים רק כדי להגיע למונופול הנוכחי שלהם. נחלק את זה במספר השנים כל מסגרת זמן שהיא - ואת מספר משתמשי ווינדוס. איך שלא תחשב את זה, זה תמיד יצא אלפים רבים של דולרים אמריקאים לשנת עבודה ממוצעת. לא שילמת כל כך הרבה כסף? ובכן, בטח שכן זה רק נחבא ממך. כן, מדינות מפותחות מבזבזות ביליונים בכל שנה על תוכנה.

כמה עלתה לינוקס? בקושי כלום. מספר המשתמשים נמוך בהרבה, אבל תצטרך להתאמץ כדי לקבל תוצאה של \$10 בשנה.

בכל זאת לפי דעתי, העלות הכללית אינה משנה במיוחד. מה שמשנה זה מה שאני מקבל בתמורה לכסף שלי. צריך לחשב את הערך הכולל של הבעלות על המוצר, ואז לחסר אותו מהעלות הכוללת הסופית של הבעלות עליו בכדי לחשב את ההטבה שמגיעה כתוצאה מהבעלות על המוצר.

אני מניח שרואי חשבון מדברים רק על עלות כוללת סופית של בעלות על התוכנה כערך "חיוני לעשיית עסקים" ולכן מדלגים על על המשמעות של הערך הסופי וההטבה שמגיע כתוצאה מהשימוש בה. אין כל ערך בתוכנה המסחרית הנורמלית, היא חיונית רק לעשיית עסקים בימים אלו. ובכן, לינוקס עונה על הצרכי המיחשוב שלי בעלות אפס, ובנוסף ההנאה האישית והערך הלימודי הם רבים.

0.15 לינוקס חולמת בהקיץ; עסקים זה מה ששולט בעולם היום

חשוב על לינוקס כעל קונסורטיום. עסקים/יחידים נאספים ביחד כדי להשיג צרכי מיחשוב דומים או כדי לפתור בעייה משותפת. ייתכן ואין להם כסף, כח אדם או מנהיג טכני אבל הם יכולים לשתף פעולה כדי לגרום ללינוקס לענות על צרכיהם. הפתרון הוא לגמרי שלהם, והוא לא צריך לעלות הרבה--הם יכולים להשתמש מחדש בחלקי התוכנה הקיימים. הם יכולים לשתף פעולה כדי לגבור על הייתרון שיש ל"מנהיג תעשייה" גדול ושיכול לשמש כנגד האינטרסים שלהם בסופו של דבר.

לינוקס זה התוצר הסופי של פעילויות שונות והרבה קונסוטוריום כאלו ש"גירדו את צרכיהם". אז לינוקס זה עסקים, אבל זה לא בהכרח עוסק במכירת תוכנה--זה יותר נוגע בגישה לתוכנה במחיר הגיוני שמתאימה לצרכיך, פותרת את הבעייה שלך, מוכרת את החומרה שלך או את השירות שלך, ושליך באופן טוטאלי (הרשיון לעולם לא פג, ולעולם לא תנושל או תורחק מקוד המקור).

0.16 לינוקס זה חרטא וכו'

אז אל תשתמש בלינוקס. אתה לא עושה לאף אחד טובה על ידי שימוש בלינוקס. GNU/לינוקס זה תוכנה חופשית ועוצמתית, אבל רק בשביל אלו שצריכים אותה. ישנן מערכות הפעלה חלופיות שאתה יכול לבחור ושיכולות לענות טוב יותר על צרכיך. למרות שרוב הלינוקסאים נהנים מהגדילה ומקבלים בברכה משתמשים חדשים, חלק לא שמחים כל כך מהעניין בגלל ש"הקהל והמסחור יכולים לקלקל את גן העדן להאקרים שיצרנו" לכן אינך עושה טובה לאיש על ידי שימוש בלינוקס. בהקשר זה, זה יכול להועיל אם נסכם את העוצמות והחולשות של לינוקס:

לינוקס נמצאת בבעלות האוהדים שלה (החלק שלך בבעלות מגיע חנם ביחד עם ההצטרפות החופשית שלך למועדון האוהדים) היא בהחלט עוצמתית ועשירה בפיצ'רים, בעל יכולות קינפוג גבוהות, גמישה עד כמה שתרצה תהיה באה ביחד עם מורכבות, הוצאת פחותות על חומרה, מגיעה עם כל יכולות הרשת המוכרות לאדם, דורשת מנהל רשת שידע קרוא וכתוב במחשבים, חלק מאפליקציות שולחן העבודה עדיין בפיגור אחרי המקבילות המסחריות בפלטפורמות אחרות, מספר רחב של אפליקציות משתמש קצה באות כסטנדרט וחינם עם מערכת ההפעלה, הממשק הגרפי נחמד מאוד אבל עדיין לא משוייף כמו של אפל או מייקרוסופט(בדוק את RH 8.0-א.א.), לינוקס עובדת בסטנדרט גבוה (יוניקס, פוסיקס), פורמטי קבצים פתוחים נמצאים בשימוש כל הזמן, אלפי תוכניות זמינות להורדה חינם (למרות שקלות השימוש והאיכות שלהן משתנות מאוד), לינוקס מהנה!

הבא:לפני התקנת לינוקס.

חלק 1. לפני התקנת לינוקס

תוכן החלק הזה:

- 1.1 באיזו הפצת לינוקס כדאי לי להשתמש? [עמ' 13]
- 1.2 מה הן דרישות החומרה של לינוקס? [עמ' 16]
- 1.3 האם החומרה שלי תעבוד תחת לינוקס? [עמ' 18]
- 1.4 איך אני מוריד את לינוקס מהאינטרנט? [עמ' 20]
- 1.5 איך אני משיג דיסק של לינוקס? [עמ' 20]
- 1.6 יש לי דיסק של רד האט אבל אין לי את הפלופי של ההתקנה [עמ' 20]
- 1.7 מה אני צריך לקרוא לפני ההתקנה? [עמ' 21]
- 1.8 האם אני יכול להתקין מייקרוסופט ווינדוס ולינוקס על אותו המחשב? [עמ' 22]
- 1.9 איך אני מחלק למחיצות את הדיסק הקשיח שלי? [עמ' 22]
- 1.10 המחיצה של מייקרוסופט ווינדוס תופסת את כל הדיסק הקשיח שלי. האם אני יכול לכווץ/לחלק את? הדיסק בלי להתקין מחדש? [עמ' 28]
- 1.11 איך אני מתחיל את ההתקנה? [עמ' 28]
- 1.12 האם ההתקנה של לינוקס קשה? [עמ' 29]
- 1.13 אילו חבילות עליי להתקין? [עמ' 29]
- 1.14 איזה ממשק גרפי להתקין, קיי.די. אי או גנום? [עמ' 30]
- 1.15 סיימתי את ההתקנה, איך אני נכנס למערכת בפעם הראשונה? [עמ' 31]
- 1.16? איך אני גורם ללינוקס לקרוס? [עמ' 31]
- 1.17 האם אני יכול להשתמש בממשק הגרפי כל הזמן? [עמ' 32]
- 1.18 איך אני משדרג הפצת לינוקס? [עמ' 33]

1.1? באיזו הפצת לינוקס עליי להשתמש

הפצת לינוקס היא אוסף קוהרנטי של תוכנות חופשיות כשהליבה של לינוקס (מערכת ההפעלה) משמשת כמרכזן.

ההבדלים בין הפצות לינוקס השונות, מינוריים: תוכנית ההתקנה, מבחר הכלים/תוכנות שמחוברים אל המערכת, אירגון כמה דברים על הדיסק הקשיח (רובו של המיקום הוא די זהה, מיקום סטנדרטי בדיסק הקשיח בכל הפצות). בכל הפצה שלא תבחר להתקין בסוף תישאר עם אותה לינוקס .
אנחנו משתמשים בעיקר ב "MANDRAKE"- (MDK) וב "RED HAT" (RH) מהסיבות הבאות:

1. שתיהן מאוד פופולריות (עובדה שגם עוזרת למשתמש חדש וגם מהווה מדד לאיכות שלהן).

2. שתיהן הפצות שמתאימות למטרות כלליות.

3. שתיהן מגיעות עם תוכניות התקנה קלות יחסית.

4. גם מנדרייק וגם רד האט הינן "תוכנות פתוחות" זה אומר שכל התוכנה שנכתבה על ידי הייצרנים ושכלולה בתקליטורי ההפצה נמצאת באופן חוקי תחת הרשיון הכללי הציבורי (ה-GPL), כך שאפשר להעתיק שלהן את התקליטורים אפשר לתת אותם, להשתמש מחדש וכו'.

5. את מנדרייק ואת רד האט ניתן להשיג בזול מאוד או בחינם אם אין לך עניין בתמיכה מסחרית. וזה. כתוצאה מ-4.

6. מנדרייק מבוססת על רד האט, כך ששתיהן דומות. חבילות תוכנה לרד האט בדרך כלל עובדות על מנדרייק ללא בעיות ולהיפך. בכל אופן, מנדרייק קצת יותר אוטומטית, בעלת שולחן עבודה נחמד יותר ודורשת פנטיום כברירת מחדל, רד האט תעבוד גם על i386 .

בקצרה, כמתחיל, אתה יכול להמר בביטחה על מנדרייק או רד האט אלא אם כן אתה מעדיף משהו אחר או שיש לך דרישות מיוחדות, או שהסביבה שלך מציעה לך להשתמש במשהו אחר(אם יש לך גורו מנוסה באזור , או כמה חברים שמתמשים בלינוקס, יכול להיות שתרצה להשתמש באותה הפצה כמוהם זה הופך את השגת העזרה למשהו קל בהרבה).

ההפצות האחרונות שאנחנו ממליצים עליהן (יוני 2000) הן רד האט 7.3 או מנדרייק 8.2. אלו הן הפצות מצויינות, הקפד לציין את ההפצה הכי עדכנית כשאתה מזמין את התוכנה שלך מסוחר-- סוחרים רבים אוהבים לנקות את המלאי שלהם על ידי שליחת הפצות ישנות יותר (וזה ישים לא רק על לינוקס). באופן כללי, הפיתוח תחת לינוקס הוא מהיר, ולא תרצה לבזבז את זמנך על הפצות ישנות, . לכותבים של מדריך. זה אין כל קשר לרד האט, מנדרייק (ואו לכל הפצה אחרת) בכל אופן שהוא.

ההמלצה שלנו על מנדרייק ורד האט למתחילים לא אומרת שהפצות אחרות לא מציעות ייתרונות או תכונות מיוחדות שעולות על רד האט או מנדרייק בתחומים מסויימים. אנחנו מאמינים שאנחנו מרוויחים מחשיפה להפצות אחרות כי זה עוזר לנו להבין את לינוקס באופן טוב יותר.

ניסינו את דביאן (DEBIAN) ואהבנו אותה מאוד. היא הייתה קלה כמו רד האט אבל נראה שהיא פחות נפוצה (לכן, בהיותנו מתחילים בחרנו ברד האט). מה שיפה בדביאן זה שהיא 100% לא מסחרית (היא מורכבת על ידי האקרים מתנדבים, בהתאם לאופי האמיתי של לינוקס) וכנראה גם הדבקה ביותר בסטנדרטים של לינוקס וכנראה גם קובעת את הסטנדרטים של המערכת.

עוד דבר יפה בדביאן זה שתקליטורי ההפצה שלה מפוצצים בכלים ואפליקציות--הרבה יותר מכל הפצה אחרת. כל האפליקציות/כלים האלה ארוזים באופן נח (כדי להקל על ההתקנה) ונבחרים לתאימות. אז כן, לא תהיה לנו כל בעיה להמליץ על דביאן הפצת לינוקס גדולה, יותר למטרה כללית. דביאן קוראת לעצמה "מערכת ההפעלה האוניבסלית" בגלל סיבה טובה.

בכל פעם דביאן מוציאה שלוש גרסאות :

1. הגירסא ה"יציבה" (לפעמים נקראת -POTATO) ולא היינו ממליצים עליה, אלא אם כן אתה באמת פאראנוייד לגבי יציבות ולא אכפת לך מחבילות די מיושנות.

2. גירסת המבחן של דביאן (לעיתים נקראת ו-WOODY) והיא יציבה בטח כמו הגירסה היציבה האחרונה של רד האט, ויותר יציבה מהמנדרייק הנוכחית שלך. היא הרבה יותר מעודכנת מדביאן היציבה. דביאן וודי היא הגירסא שאנחנו אוהבים.

3. אם לא אכפת לך להיתקל קצת בצרות, אתה יכול לנסות גם את הוורסיה השלישית שנקראת "לא יציבה" (unstable) אבל קרוב לוודאי גם הכי מעודכנת.

קורל (COREL) עבדו פעם על גירסאת לינוקס משלהם. כנראה בכדי ליצור פלטפורמה נוחה עליה יוכלו להריץ את חבילת יישומי האופיס שלהם: וורדפרפקט, וורדפרוססור, גליון הנתונים קוואנטום פרו, את תוכנת המצגות של קורל, מסד הנתונים פרדוקס, חבילת האומן של קורל דרו... קורל לינוקס התבססה על דביאן, היא דווקא נראתה די מבטיחה, אבל לא ברור לי מה קורל עשתה איתה (קיבלה כסף ממיקרוסופט?) כדי לרדת מהנושא.

בקיצור, קורל לינוקס מתה עכשיו, ולעולם לא אמליץ עליה לאף אחד מכיוון שהיא דרך ללא מוצא. הסיבה היחידה להזכיר אותה כאן היא בגלל שקורל לינוקס זכתה פעם לפרסום רב, אז יכול להיות שעוד תשמע עליה פה ושם.

(http://www.corel.com/learning/wbt/linuxos1/-/א.א)

סלאקוור (SLACKWAR)

חביבה בדרך כלל על האקרים מהשורה הראשונה שאוהבים להיות קרובאל מערכת ההפעלה ואל החומרה שלהם--לעולם לא השתמשנו בה אבל זה מה ששמענו.

יהיה לנו קשה להמליץ על סלאקוור למשתמשי לינוקס חדשים
הכתב שלנו ביל סטילס מדווח: "הסיבה האמיתית שבגללה משתמש חדש צריך להמנע
מסלאקוור היא בגלל שהיא דורשת שימוש רב יותר בשורת הפקודה ובגלל שהיא חסרה את
הכלים החלקלקים והנוטפים שיש להפצות אחרות." בכל מקרה קיבלנו משוב ממשתמשי
לינוקס חדשים שמשתמשים בסלאקוור והיא עובדת טוב מאוד אצלם. לכן עם קצת מאמץ,
אדמיניסטרטור שמבין במחשבים יכול להבין באמת מה קורה במערכת ההפעלה שלו (זה
משהו שאני לא תמיד יכול לומר על מנדרייק, או על מייקרוסופט ווינדוס).
אולי סלאקוור היא בשביל לינוקס מה שדוס עבור ווינדוס :

S.u.S.E-זזה

היא הפצה מאוד פופולרית באירופה. היא בהחלט נראית גרמניה--הפצה מוצקה למטרות
כלליות עם תוכנית התקנה קלה ומוניטין מצויין. משתמשים רבים נשבים בסוזה. לא יכולנו
למצוא תקליטורי סוזה זולים למרות שנראה שאפשר להוריד אותם מהרשת.

קלדרה הינה עוד הפצה ידוע ומכובדת מאוד. אומרים שהיא מכוונת אל המשתמש הארגוני,
יש לה את תוכנית ההתקנה המפוארת ביותר, אוסף של כלים מתקדמים (ויקרים) כלים
לקינפוג מרחוק, ועוד כלים טובים שמתאימים לארגון.

קלדרה רכשה את היוניקס המקורית-SCO-, באוגוסט 2000, (מה שמשווה לה בעיניי מראה
עוד יותר "אירגוני" ומקצועי) וגם את הסמל המסחרי-UNIX- נראה שקלדרה לא משקיעה
חלק גדול מהעבודה שלה בקהילת הלינוקס, וגם לא נראה שאכפת לה ממשתמשי לינוקס
הביתיים, אז נראה לי שלא אשקול שימוש ביתי במערכת הזו.

ישנן גם גרסאות מקומיות ללינוקס למדינות מסוימות או שפות (קוראנית, סינית, יפנית...)
והן בדרך כלל מכילות (כברירת מחדל) את כל התיעוד שמשתמשים במדינות האלה רוצים
לראות.

ביל סטיהל מספר: "ייתכן שתמצאו להזכיר את ההפצה הברזילאית קונקטיבה, באופן כללי
היא מבוססת על רד האט והיא זמינה בפורטוגזית ובספרדית".

בדוק ב <http://www.conectiva.com.br/>

שמעתי כמה דברים טובים על קונקטיבה, אז בטח הייתי משתמש בה אם פוטוגזית או
ספרדית היו השפות שלי.

ישנן גם הפצות לינוקס בעלות "מטרות מיחדות", מהודורות real time, (יכול להועיל אם
אתה בעניין של אוטומציה, רובוטיקה, עיבוד מידע מהיר וכו').

הפצות לינוקס קטנות מאוד (אם מוצא חן בעינייך הרעיון של הרצת לינוקס מתוך פלופי יחיד
שיכול להועיל לאבטחת המערכת או להתאוששות מערכת).

לינוקס למערכות מוטמעות (אם אתה רוצה לעצב את לינוקס למכשיר קטן ובעל מטרה
"מיוחדת" או לטלפון סלולרי מרשים).

מיחשוב מקבילי ומערכות של אשכולות מחשוב (יכול להיות נהדר אם אתה מתכנן מערך חיזוי מזג אוויר משל עצמך, או מדמה פיצוצים גרעיניים למשל...).

במקרים האלו ההבדלים יהיו גדולים יותר, אבל הפצות אלו לא נבנו ל"מטרות כלליות" אני מאמין שכמשתמש חדש, תעדיף שלא להשתמש באחת מאלו, למרות שיכול להיות שתיתפתה לכך.

הן ללא ספק מראות את העוצמה ואת השימושיות של לינוקס--לינוקס רצה על צעצועים, אפילו על שעון יד, כמו גם על אשכולות מיחשוב שבונים את המערכות הכי מהירות בעולם, נכון להיום.

ההפצה שאתה צריך חייבת להיות מותאמת לפלטפורמת החומרה שיש לך. זה אומר שאם יש לך חומרה שמכילה מעבד אינטל 386, או אינטל 486, או אינטל פנטיום, או אינטל 586, או אינטל 686, או סירקס, או קיי6, איי.אמ.די. או משהו דומה, אתה צריך את ההפצה הבינארית שנקראת "אינטל" או "386" "x86", אלא אם כן אתה מתכנן להכין אוסף הפצות מקור של לינוקס, (מה שבדרך כלל לא טיפוסית למשתמש חדש).

זה קורה כי ישנן הפצות בינריות לפלטפורמות חומרה אחרות פאורפיסי, אלפה, אפל, מחשבי מיינפריים של איי.בי.אם, אינטל סטרוג ארם, טרנסמטה, וכנראה גם לעוד תריסר-- אתה לא תרצה להשיג את הבינארים האלו לתואם פיסי שלך: הם בטוח לא יעבדו על פיסי עם מעבד אינטל או איי.אמ.די בפנים.

אם יש לך חומרה שהיא לא אינטל, יכול להיות שתרצה לחפש באינטרנט כדי למצוא מי תומך בה.

רוב הסיכויים שדבין תומכת בה, נראה שהיא תומכת בחומרה האקזוטית ביותר, (אתה תצטרך להזמין, דביןחזק או דבין מוטורולה 0*686, או דבין פאור פיסי, או דבין ספארק....).

בקיזור, אף על פי שמשתמשים חדשים מתבלבלים מכל ההפצות הרבות של לינוקס, ישנן מספר סיבות להחזקת הפצות שונות. צריך להסתכל עליהן כעל העוצמה של לינוקס ולא כעל חולשתה. לינוקס פשוט ממלאת את כל הנישות של האפליקציה והחומרה.

המדריך הזה מתרכז ברד האט ומנדרייק לפלפורמת הפיסי (אינטל). רבות מהתשובות יישמות גם לגבי הפצות אחרות או פלטפורמות אחרות, אבל לא ניסינו אותם.

באילו הפצות לבחור בשביל החומרה הישנה שלי? תשובה מהירה: דבין, סלייקוואר, או אולי בייסיק לינוקס (הגירסה העדכנית), או אולי גירסה ישנה של רד האט, מנדרייק או סוזה. ההצדקה: רד האט, מנדרייק, סוזה, קלדרה וטורבו לינוקס מותאמות להוציא את המקסימום מהחומרה העדכנית ביותר ליום הוצאתן. יכול להיות שזה יהיה קשה או בלתי אפשרי להתקין אותן על מכונת ישנות לרוב בגלל מגבלות של זכרון ומהירות. דבין וסלייקוואר מתאימות גם הן לרוב החומרה הישנה.

1.2 מה הן דרישות החומרה של לינוקס?

לינוקס מהקופסה אמורה לרוץ על פיסי מבוסס 386 עם 8 מגה זיכרון, אבל כזה מחשב מתאים רק לאפליקציות מבוססות טקסט בלבד (בלי X-windows).

486 עם 16 מגה זיכרון ו-600 מגה פנויים (לא מחולקים למחיצות) על הדיסק הקשיח יעבוד תחת איקס ווינדוס אבל אל תצפה ממנו שיטוס.

ה 586 שלי עם 133 מגה הרץ ו 64 מגה זיכרון רץ בצורה סבירה תחת לינוקס עם איקס. האתלון 1.3 ג'יגה שלי עם 256 מגה זיכרון הוא תענוג אמיתי, עם תגובה מיידית אפילו כשרצות עליו כמה אפליקציות בבת אחת. היום לא אקנה מחשב עם פחות מ-256 מגה זיכרון (דצמבר 2001).

ל-486 עם 33 מגה הרץ, 8 מגה זיכרון ודיסק של ג'יגה אחד יש מעט מדי זיכרון כדי לתפקד כראוי תחת משק גרפי, אבל הוא עדיין יעיל ברשת הפנימית שלי בתור טרמינל של איקס.

מכונה עם 486 מתפקדת כמו שצריך ובזכות עצמה רק אם יש לה לפחות 16 מגה זיכרון (אבל לפעמים קשה להשיג זכרון למחשבים ישנים במחיר שאתה חושב שהוא שווה). הנייד הישן שלי, טושיבה 386 אס.איקס עם מעבד 20 מגה הרץ, 9 מגה זיכרון ו-120 מגה דיסק קשיח מריץ "אפליקציות legacy" תחת ווינדוס 3.11, מתחבר לרשת הביתית שלנו ולכן עדיין יעיל. ניסינו על הטושיבה הישן הזה גם לינוקס דבין, והוא רץ מצויי נבמצב טקסט. (בבקשה שים לב שמנדרייק דורשת מעבד פנטיום).

אם אתה מעוניין לקפוץ כמה משוחות, אתה אמור להצליח להתקין ולהריץ לינוקס על מחשב עם 4 מגה זיכרון, אבל זה בטח לא שווה את המאמץ בשביל מכונת לינוקס ביתית. לפי דעתי: 32 מגה זכרון, ואם זה אפשרי 128 או 256 מגה. יותר זכרון יכול לגרום להבדל בביצועים כשמריצים כמה אפליקציות ממשק גרפי גדולות ביחד. הזכרונות זולים היום. שים לב בבקשה שלהפצות רבות יש בעיות להריץ את תוכניות ההתקנה שלהן על מחשבים ישנים עם מעט זכרון (למרות שברגע שהיא הותקנה, לינוקס בדרך כלל תרוץ בסדר).

אם אתה זקוק לעוד עזרה לגבי התקנת לינוקס על מחשבים עם מעט זכרון, נסה את:

<http://7thguard.net/files/debianHOWTO.TXT>

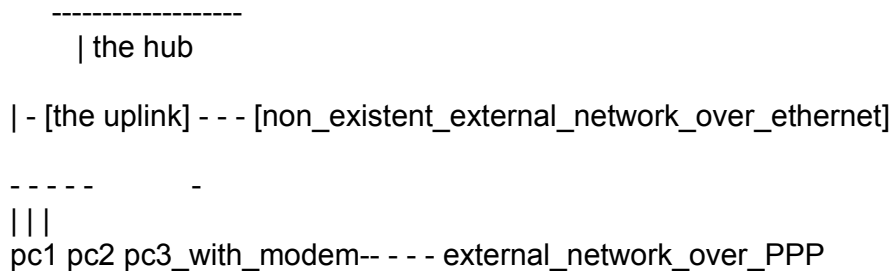
רשתות זה התחום שבו לינוקס באמת זורחת, אז תתחיל לחשוב על להשיג 10 כרטיסי רשת--הם לא יקרם מדי ויהיו מושלמים כדי לחבר ביחד את המחשבים שיש לך בבית. בנוסף נסה להשיג כרטיסי רשת ישנים שנחשבים למושנים מדי תחת מייקרוסופט ווינדוס. ניתן לקנות אותם במחיר מאוד נמוך הם יעבדו טוב מאוד תחת לינוקס.

כדי לחבר שני מחשבים בלבד, כבל רשת מוצלב לחיבור ישיר יכול להספיק ("רשת לעניים"). כדי לחבר יותר משני מחשבים ביחד, תזדקק להאב (בין 30 ל-80 \$) וכבלים רגילים (לא מוצלבים). אם יש לך כרטיסי רשת מיותרים, אתה יכול להתקין יותר מכרטיס אחד על מחשב, השתמש בכלים מוצלבים, ותחסוך את ההוצאה על ההאב, אבל זה דורש דורש קינפוג מורכב יותר של המערכת.

מערכת-T-BASE 10 משתמשת במחברי רשת מסוג טלפון ענק (RJ45).

כל המכונות מחוברות לקופסה אחת שנקראת האב HUB.
להאב יש חיבור נוסף (שנקרא אפלינק - UPLINK) שיכנס לשימוש אם אי פעם יהיה לי
חיבור קבוע אחרי הרשת הפנימית לעולם החיצון.

הנה סידור סכמטי פשוט של רשת ביתית:



כאן אני מראה רשת מקומית פרטית שציגה את פיסה 1,2 ופיסי 3, מחוברים דרך האב. מכיון
שלא חובר אצלי לאפלינק חיבור לרשת חיצונית, פיסה 3 מספק את התקשורת שלי לעולם
החיצון (דרך מודם). לכן פיסה 3 נקרא- שער-GATEWAY- לכל המחשבים ברשת המקומית
הפרטית שלי (פרט לפיסה 3 בעצמו). איפשרתי את תוכנת הפייר וול ב PPP של התקן הרשת
בפיסי 3, וגרמתי לפיסי 3 לדעת איך 3 לחייג החוצה. העולם החיצון יכול לראות רק את פיסה
3 מבחינתם פיסה 1 ו 2 לא קיימים. הרשת. המקומית שלי בטוחה כי רק לאנשים מורשים יש
גישה פיסית לפיסי 1 ו 2 ופיסי 3.

PPP מתפרש כ POINT TO POINT PROTOCOL, והוא משמש לתקשורת על גבי קווי
טלפון.

כמובן שאתה יכול לבנות רשת הרבה יותר מורכבת עם לינוקס. למחשב יכולים להיות שני
כרטיסי רשת או יותר, והוא יכול לשמש כגשר בין שתי רשתות או יותר. המחשב ישמש
כשער לכל התנועה בין שתי הרשתות. הרשתות לא חייבות להיות מוכרות לעולם החיצון
("רשת מקומית פרטית") ולשבת מאחורי פייר וול שמאפשר על המחשב שמשמש כשער.
העולם החיצון יידע רק על מחשב אחד, השער לרשת החיצונית. מחשבים אחרים עדיין
יכולו לתקשר עם העולם החיצון, אבל כל התנועה תראה כאילו היא מגיעה ממחשב עמוס
אחד-השער.

1.3 האם החומרה שלי תעבוד תחת לינוקס?

לא כל החומרה עובדת תחת לינוקס, למרות שרוב הציווד כן יעבוד תחת לינוקס, במיוחד
חומרה, סטנדרטית, פופולרית או ישנה. זה נכון גם לגבי מתאמי סקאזי, סידי רום, צורבים,
כרטיסי ווידאו, מיקרופונים, מדפסות, מודמים, כרטיסי רשת, סורקים, כונני זיפ וכו'.

יוצאי הדופן הם מודמי התוכנה ה WINMODEMS הימנע מהם כמו מאש הם קצת יותר
זולים מודמים מלאים, אבל הם גם קצת נכים (חלק מעבודת העיבוד מתבצעת על ידי המעבד
של המחשב במקום על ידי המודם עצמו), ויש סיכוי נמוך שתצליח להפעיל מודם כזה עם
לינוקס מיד עם ההתקנה.
הסתכל ב: <http://www.idir.net/~gromtikc/winmodem.html> למידע נוסף.

מודמים חיצוניים אינם ווינמודם, הם קצת יותר יקרים אבל לא גוזלים מהמחשב שלך חשמל ומשאבים הם מתניידים בקלות בין מחשבים ומראים את פעילות המודם .
נקודות נוספות בנוגע למודמים:
"מודמים חיצוניים ישנים שמשתמשים בפרורטוקול של רוקוול לא עובדים כל כך טוב. בנוסף , מודמים בעלי חיבור USB החדשים יותר, אינם נתמכים כרגע(מרץ 2001) בצורה טובה."
מקור-B. Staelhe .

תחום נוסף עם בעיות פוטנציאליות הוא כרטיס הווידאו. אם יש לך כרטיס חדיש מאוד עם תלת מימד או כרטיס לא קונבנציונלי, אתה יכול לבדוק את התאימות שלו ב-
<http://www.xfree86.org>.

כונני זיפ מכל הסוגים נתמכים היטב.

לא הייתי סומך על לינוקס שתתמוך סורק בעל חיבור מקבילי, ולא משנה אם הייצרן טוען לתאימות (TWIN=(Technology Without An Interesting Name - no joke here)).

התשובה הקצרה היא כן, מרוב הבחינות הפיסיסטנדרטי שלך יריץ לינוקס בלי בעיות. אתה לא משקיע הרבה כשאתה מנסה את לינוקס, לכן הדרך הקלה ביותר הלבדוק היא להתקין לינוקס על החומרה: הנוכחית שלך. ישנן רשימות תאימות חומרה ב:

<http://hardware.redhat.com/hcl/genpage2/cgi>
<http://metalab.unc.edu/ldp/howto/hardware-howto.html>

אם אתה רוצה לבדוק את החומרה שלך.

עם רכישת חומרה חדשה, אני תמיד בודק את התאימות שלה ללינוקס ברשימות הנזכרות למעלה. אתה תמיד יכול לשאול את ספק החומרה שלך אם החומרה נתמכת תחת לינוקס, אבל הייתי לוקח את התשובה. בערבון מוגבל, לחברות רבות יש תמיכה טכנית/אנשי מכירות שלא מוכשרים לתפקיד.

הייתי שוקל לקנות מחשב שלינוקס כבר מותקנת עליו.

אם אתה אדם הרפתקני, כמוני, הייתי מתעלם מן ההערות האלו, הסיכויים הם 10-90 שהחומרה תעבוד.

אם חומרה שבבעלותך לא נתמכת תחת הפצת לינוקס הנוכחית, אל תרים ידיים. הסיכויים הם:

1. היא כן נתמכת, אבל אתה לא יודע איך להתקין אותה (פתרון: השאר עם לינוקס לכמה שבועות, אל תבזבז את הזמן, כשתהיה לך קצת הבנה לגבי איך המערכת שלך עובדת, אולי תוכל להתקין אותה).

2. אתה צריך לעבור תהליך התקנה מורכב יותר כדי לתמוך בחומרה (לתת איזו פקודה שאתה לא מכיר או לקמפל מחדש את הקרנל, מה שברך כלל לא קשה כמו שזה נראה).

3. הפצה מעודכת יותר (אחרת ??) תומכת בחומרה מהקופסה

4. כבר קיים שידרוג איפה שהוא באינטרנט תצטרך למצוא אותו, להוריד אותו, ולהבין איך להתקין אותו.

1.4 איך אני מוריד לינוקס?

עשה לעצמך טובה ואל תוריד את לינוקס מהרשת. קנה במקום זה דיסק התקנה. ניתן להוריד את לינוקס בשלמותה מהאינטרנט, אבל זוהי מערכת הפעלה גדולה מאוד ומתוחכמת מאוד. ההורדה יכולה לקחת שעות או ימים, ואתה יכול להיתקל בבעיות ותיסכולים בגלל טעויות בקבצים שאתה מוריד אם יש לך חיבור מהיר לאינטרנט (בהחלט לא מודם, 56 אולי מודם כבלים) ואינך מתשתמש חדש מאוד הורדת לינוקס יכולה להיות אופציה.

נסה ב: <http://www.linuxiso.org/> לאימג'ים מוכנים לצריבה של הפצת לינוקס המועדפת עליך.

1.5 איך אני משיג דיסק של לינוקס?

(חפש קצת בפורומים הישראליים של לינוקס ובאינטרנט-אם יהיה לך רצון, אתה תמצא-א.א.)

1.6 יש לי את הדיסקים להתקנת לינוקס אבל אין לי את הפלופי להתקנה. מה אני עושה?

אם המחשב שלך תומך באיתחול מכונן התקליטורים (רוב המחשבים הישנים אינם תומכים) אינך זקוק לפלופי לאיתחול ההתקנה של לינוקס. בדוק את הביוס שלך: ניתן לקבוע את סדר האיתחול משם בדרך כלל. במחשב שלי כונן התקליטורים הוא מכשיר האיתחול הראשון בביוס ועדיין איני יכול לאתחל מהכונן הזה. ככה שהגדרת הביוס לא בהכרח משקפת את היכולות של המחשב שלך, אם אתה יכול לאתחל מכונן התקליטורים, כל מה שעלייך לעשות בכדי להתחיל את תוכנית ההתקנה זה להכניס את הדיסק של רד האט אל הכונן ולאתחל את המחשב. במידה והמחשב שלך לא מאתחל מכונן התקליטורים, הכן דיסק התקנה תחת dos.

חשוב מאוד שיהיה לך פלופי תקין (ללא סקטור פגום אחד). התוכנית שמכינה את הפלופי לא בודקת אם הוא נכתב כראוי. בנוסף, אל תסמוך על תוכנת הפורמט של דוס שתמצא פלופי לא שמיש--סביר להניח שהיא לא. אם אני הייתי אתה, הייתי מניח שניים או שלושה פלופים בבת אחת--תופתע לדעת לכמה דיסקטים יש בעיות. אצלי הפלופי השלישי עבד! אם דיסקט ההתקנה שלך לא מאתחל, הכן אחד נוסף--הוא אמור לעבוד.

הנה הפקודות. כדי להריץ את הפלופי:

```
F:\dosutils\rawrite.exe -f F:\images\supp.img -d a: -n
```

כדי להריץ את הדיסק הנוסף (אופציונלי):

```
F:\dosutils\rawrite.exe -f F:\images\supp.img -d a: -n
```

בהנחה שכונן התקליטורים שלך תחת דוס הוא: F, והפלופי הוא: A. התאם את הפקודות במידה והכוננים מופיעים אחרת במערכת שלך.

20 הפקודות למעלה מריצות את היטיליטי שנקרא "rawrite" ומציינות את מיקום קובץ הקלט, תעתיק הדיסק תחת האות f, את הכונן המיועד אחרי האופציה d ועל ידי האופציה n מורה להכניס פלופי כיעד.

יכול להיות שתמצא שקל יותר להריץ את rawrite- ללא כל ארגומנטים--אתה תתבקש לקבוע את האימאג' ואת אות הכונן המיועד.

תחת לינוקס, אתה יכול להכין פלופי על ידי המרת הדיסק של רד האט והקלדת הפקודות הבאות כROOT:

```
cd /mnt/cdrom/images/
```

```
dd if=boot.img of=/dev/fd0
```

הפקודה למעלה מציינת ש: קובץ הקלט if הוא boot.img. ושקובץ הפלט "of" הוא /dev/fd0 שהוא למעשה הפלופי. הפקודה "dd" מעתיקה קבצים.

אם יש לך כונן פלופי נוסף ואתה רוצה לכתוב אליו השתמש ב1 בסוף ולא ב0.

1.7 מה עליי לקרוא לפני ההתקנה?

עוזר מאוד לקבל קצת אוריינטציה יוניקס אם אין לך בכלל. קנה מדריך לינוקס טוב או שתסיר את האבק ממדריך היוניקס הישן שלך. כמעט כל העקרונות של יוניקס מוחלים על לינוקס, וכמעט כל הפקודות של יוניקס ירצו טוב תחת לינוקס. אני מוצא את המדריכים למייקרוסופט ווינדוס כחסרי תועלת (תקליק כאן, תקליק שם, הסתכל בצילומי המסך), לעומת זאת מדריכי לינוקס/יוניקס בדרך כלל טובים (מקנים לך הבנה לגבי המערכת, וזה ילווה אותך לאורך זמן).

יכול מאוד להיות שתמצא ללמוד על החומרה שלך: כמה ומה גודל הדיסק הקשיח שלך, הסוג, המספר סדר וגודל כל המחיצות בכל כונן, היכן ממקומות מחיצות ה דוס/ווינדוס שלך, מי מהן היא מחיצת האיתחול של דוס/ווינדוס (אם אתה מתכנן איתחול כפול) מהו סוג העכבר שלך, איזה כרטיס מסך יש לך וכמה זיכרון יש לו, איזה מסך (תדירויות סינכרון מקסימליות) וכו'.

גש אל הביוס ובדוק את המספר והגאומטריה של הדיסק הקשיח שלך. הרץ את fdisk תחת דוס כדי להציג את טבלת המחיצות שעל הדיסק, והדפס אותה. התבונן באיתחול המערכת שלך כדי ללמוד על סוג כרטיס המסך שלך ועל כמות הזיכרון שלו. אתחל את מייקרוסופט ווינדוס, גש אל מנהל ההתקנים בלוח הבקרה ורשום לעצמך איזה כרטיס קול, מודם, כרטיס רשת יש לך ומה ההגדרות שלהם (שם, סוג, irq, כתובות קלט/פלט, ערוץ dma, קרא מה רשום על התוית מתחת לעכבר שלך כדי לבדוק מה סוג העכבר שלך) (בפעם הבאה שתקנה עכבר, קנה לוג'יטק עם 3 כפתורים או משהו דומה -- לינוקס עושה שימוש טוב בכל שלושת הכפתורים) בדוק בחוברת שהגיעה עם המסך שלך מה הם תדירויות הסינכרון המקסימליות (אנכי ואופקי) שבהן המסך שלך תומך.

לעולם אל תשתמש בתדירויות שחורגות מעבר ליכולות המסך שלך -- זה עלול לפגוע בו.

ייתכן שתמצא לדפדף קצת בחבורות של רד האט או של מנדרייק. אם אין ברשותך עותק מודפס, גירסת html נמצאת על הדיסק שלך אז את היכול לנסות לקרוא אותה על יד דפדפן.

1.8 האם אני יכול להתקין מייקרוסופט ווינדוס ולינוקס על אותו מחשב?

כן אתה יכול. לינוקסאים רבים משתמשים באיתחול כפול. ניתן לעשות את זה על ידי התקנת ווינדוס על מחיצה אחת והתקנת לינוקס על אחרת.

לינוקס מגיעה עם מנהל איתחול פשוט בשם LILO או אחד מתחוכם יותר בשם GRUB שיאפשרו לך לבחור, בזמן האיתחול את מערכת ההפעלה שאתה מעוניין לאתחל. קודם כל התקן את מייקרוסופט ווינדוס ואחר כך את לינוקס אחרת ההתקנה של ווינדוס תבטל את הגישה שלך ללינוקס. תכין לעצמך דיסקט איתחול למקרה שתצטרך להתקין את ווינדוס מחדש--ווינדוס בטוח תבטל את הגישה שלך ללינוקס, אתה תצטרך לאתחל את לינוקס מהדיסקט ולהריץ שוב את הפקודה lilo כדי לחזור לעינינים.

תחת לינוקס, אתה תוכל לקרוא ולכתוב למחיצות של ווינדוס כך שהעברת הנתונים בין ווינדוס ללינוקס הינה שקופה. בנוסף תוכל גם להשתמש במשאבים מבוססי ווינדוס: קבצי קול, רקעים, תמונות, פונטים וכו' (רצוי שתבדוק לפני כן שאינך מפר את הסכם הרישוי, עבור מוצרים שאתה שוכר ממיקרוסופט סביר להניח שאתה כן. עם לינוקס, אני גאה בעובדה שאין שום תוכנה פיראטית במערכת שלי).

1.9 איך אני מחלק למחיצות את הדיסק הקשיח שלי?

לפני התקנת לינוקס, תרצה לדעת מה זה בכלל מחיצה על דיסק קשיח. הדאגה היא שתמחק את מחיצת הווינדוס שלך כשאתה למעשה לא חפץ בכך, אתה מעוניין בשתי מחיצות שונות כדי שתוכל ליישם איתחול כפול. זה אומר: מייקרוסופט ווינדוס על מחיצה אחת, לינוקס על השניה. בדרך כלל לא מתקינים לינוקס על שטח פנוי במחיצה של ווינדוס, למרות שזה אפשרי, אנחנו לא ממליצים על זה.

אם אתה מתכנן איתחול כפול ראשית השתמש ב fdiks תחת דוס כדי ליצור את המחיצות של ווינדוס, השאר חלק מהדיסק לא מחולק למחיצות בשביל לינוקס. אתה תכין ותפרמט את המחיצות בזמן התקנת רד האט (או מנדרייק או מה שלא יהיה). לינוקס תזזה את השטח הפנוי על הדיסק הקשיח. הפוך את מחיצת הווינדוס שלך למחיצת האיתחול ולראשית. התקן, קנפג ובחן את הווינדוס שלך לפני התקנת לינוקס. אם אתה מתכנן להתקין לינוקס בלבד, כל מה שאתה צריך זה דיסק קשיח נקי, ללא מחיצות כדי להתחיל.

ניתן ליצור רק מחיצת לינוקס אחת אם אתה מעוניין בכך (בנוסף למחיצת הווינדוס אם אתה משתמש באיתחול כפול), אבל כדאי שיהיו לך עוד כמה מחיצות כדי שתוכל לשמור נתונים

של משתמשים בנפרד משאר מערכת ההפעלה. ככה, אם משהו מסתבך, או אם פירמטט מחדש והתקנת מחדש את מערכת ההפעלה, אינך מאבד נתונים של משתמשים אתה יכול לבצע התקנה מחודשת של לינוקס בלי לאבד את התוכן של /home, הסיפרייה שמכילה את כל הנתונים של המשתמשים, במידה ואתה מדלג על אופציית הפירמוט מחדש. אבל כדי להשיג את זה ספריית /home צריכה להיות על מחיצה משלה.

בזמן התקנת לינוקס, אתה תתבקש לחלק למחיצות את השטח הפנוי שעל הדיסק הקשיח שלך. ישנן דרכים רבות לחלק למחיצות את הדיסק, תלוי בגודל של הדיסק הקשיח שלך, בדרישות ובהעדפה שלך אני אוהב מחיצות לינוקס כאלו: (למודל צנוע של דיסק עם שני ג'יגה שנתתי ללינוקס בדוגמא הזו):

mount point	type	size
/	ext2	300MB
/usr	ext2	1200MB
/home	ext2	380MB
swap	swap	120MB

בדוגמה זו, ייעדתי 300 מגה למחיצה שמחזיקה את הבסיס למערכת ההפעלה לינוקס, המחיצה הראשית the root partition, בנוסף הקדשתי 1200 מגה לנקודת המרה שתיראה במערכת שלי כספריית ה /usr אשר תכיל אך ורק את תוכניות ששיכות למשתמש (תוכניות שלא מגיעות עם המערכת הבסיסית ושאינן אתקין בשלב מאוחר יותר, כמו start office).

הקדשתי גם 380 מגה למחיצה שתיראה כ /home אשר תכיל את ההגדרות והנתונים לגבי כל המשתמשים במערכת. ולבסוף נתתי עוד 120 מגה שישמשו את המערכת לזיכרון וורטואלי (ההארכה של זיכרון הסיליקון הפיזי, על גבי הדיסק הקשיח, שנקרא גם swap).

אם הקרנל שלך נמוך מ-2.2 (וזה המקרה ברד האט 5.2 סטנדרטי וכל מה שלפניו), מחיצת הסוואפ שלך לא יכולה להיות גדולה מ-127 מגה. חוק האצבע הוא שמחיצת הסוואפ צריכה להיות פעמיים הזיכרון הפיזי אם אתה מעוניין ביותר (אם יש לך הרבה זיכרון פיזי, או שאתה מתכוון להריץ תוכניות עם מבני נתונים גדולים) יכול להיות שתצצה ליצור מחיצת סוואפ גדולה יותר בזמן ההתקנה (או כמה מחיצות סוואפ קטנות יותר) או להוסיף קובץ (ים) סוואפ מאוחר יותר.

שני ג'יגה זו כמות מכובדת של שטח דיסק שאמורה להספיק למשתמשים שאוהבים אפליקציות רבות זה מתאפשר בגלל שאפליקציות של לינוקס נוטות להיות רזות יותר מהמקבילות המייקרוסופטיות שלהן) בכל מקרה, אם תנסה להתקין את כל מה שזמין על גבי הדיסקים של ההפצות המודרניות, ייגמר לך המקום בדיסק הקשיח. הניסיון שלי לימד אותי שלא משנה עד כמה גדול הדיסק, הוא תמיד מתמלא ואני מתחרט על זה שאין ושלא רכשתי דיסק גדול יותר.

במידה והשטח על גבי הדיסק הקשיח שלי מוגבל אני שוקל הקמת שתי מחיצות כמו בדוגמא הזו (למחיצה דלה של 650 מגה שמיועדת ללינוקס):

mount point	type	size
/	ext2	600MB
swap	swap	50MB

כאן הקדשתי 600 מגה לבסיס של מערכת ההפעלה, אפליקציות, ולנתונים/מסמכים של המשתמש והקצתי 50 מגה למחיצת הזיכרון הווירטואלי, מחיצת הסוואפ. ה-50 מגה סוואפ אמורים להספיק לפעולות הכרחיות בעלות משקל בינוני. מגבלת ה-600 מגה למערכת ההפעלה, אפליקציות ונתוני משתמש תכריח אותך להיות בררני ביותר בבחירת האפליקציות שתתקין או שתסתכן בכך שייגמר לך במקום על הדיסק.

נסה ללחוץ על <F1> כשאתה מתקין תוכנה אופציונלית שמגיעה על גבי הדיסק של רד האט אתה תקבל תיאור קצר לגבי מה התוכנה עושה וכך תוכל להחליט אם אתה באמת זקוק לה. (אל תדאג יותר מדי אם החמצת משהו שאתה צריך, תמיד תוכל להתקין את החלקים החסרים מאוחר יותר). אתה יכול לסיים את בקלות את ההתקנה של רד האט עם 200 מגה פנויים על הדיסק הקשיח (מתוך 600 בדוגמא הזו) במידה ואתה בוחר בחירות הגיוניות. שים לב, ש"הדבקה" של מחיצת ה-root/ ושל מחיצת ה-home/ ביחד, אולי תחסוך לך מקום, אבל היא אינה הפתרון הבטוח ביותר.

ניתן להתקין לינוקס על גבי שטח דיסק קטן יותר מאשר בדוגמא למעלה, אבל תצטרך להיות בררן מאוד לגבי מה שאתה מתקין.

לדיסק בעל נפח גדול יותר, הייתי שוקל את ההתקנה הבאה (ל-15 מגה מפנקים וייעודיים ללינוקס):

mount point	type	size
/	ext2	800M
/usr	ext2	5000MB
/usr/local	ext2	3000MB
/home	ext2	5200MB
swap	swap	1000MB

אנא שים לב שנקודות ההמרה יכולות לשכון על גבי דיסקים קשיחים שונים. לינוקס אוספת את כל שטח הדיסק אל תוך אשכול אחד.

נקודה נוספת שצריך לקחת בחשבון כשמקימים מחיצות:

ביוסים ישנים רבים מגבילים את מחיצת האתחול ככה שהיא לא יכולה להמשיך מעבר לצילינדר ה-1024 על גבי הדיסק הפיזי הראשון. כדי להתגבר על המגבלה הזו, כל מה

שעלייך לעשות זה ליצור את מחיצת האיתחול הראשונה שלך כך שהיא תסתים לפני צילינדר מספר 1023 (זה גורם לכך שהמחיצה תהיה בגודל שלך 512 מגה בערך שזה המון ל /root). ברגע שלינוקס תרוץ, למגבלת הביוס כבר לא תהיה כל חשיבות מכיוון שלינוקס תשתלט על ניהול החומרה והיא דווקא כן יכולה לגשת אל מחיצות שמעבר לצילינדר מספר 1023.

כשמתקינים ומשתמשים בלינוקס, הכוננים שלך מופיעים כהתקנים בעלי השמות הבאים:

hda -- כונן ide הראשון (מתפרש כ"hard drive a", כונן המאסטר בממשק ide הראשון).

hdb -- כונן ide השני (הכוונה היא כאן לכונן ה slave בממשק ה ide הראשון).

hdc -- כונן ide השלישי (כונן המאסטר בממשק ide השני).

hdd -- כונן ide הרביעי (ה slave בממשק ה ide השני).

המספרים מצביעים על מחיצות בתוך בדיסק הקשיח:

hda1 -- המחיצה הראשונה ב hda.

hda2 -- המחיצה השנייה ב hda.

hda3 המחיצה השלישית ב hda.

וכן הלאה, אם יש לך יותר מחיצות בכונן הראשון:

hdb1 -- המחיצה הראשונה שבכונן השני או רק hdb אם הסידי רום מותקן כסלייב בממשק ה ide הראשון.

hdc1-המחיצה הראשונה בכונן השלישי, וכן הלאה.

לכונני סקאזי יש שמות מקבילים אבל הם מתחילים באותיות "sd", SCSI drive= כשבהמשך מופיע מספר שמצביע על הממשק ומספר שמצביע על ההתקן.

sda4 למשל מתפרש ככונן הסקאזי הראשון, בעל מספר הזהות 4.

במידה ויש לך כונן זיפ חיצוני שמחובר לחיבור המקבילי של המחשב שלך, הוא יופיע כהתקן סקאזי sda4- כונני זיפ עובדים במצב שמדמה סקאזי.

רשימת המחיצות שמוצגת לפניך בתוכנית ההתקנה של לינוקס תכלול את כל המחיצות של מייקרוסופט ווינדוס שיש לך. לדוגמא, לי יש את מחיצות ווינדוס הבאות:

mount point	type	size	comment
[no mount]	vfat	1200MB	["win C drive, hda1]
/mnt/dos_hdd2	vfat	1600MB	["win D drive' hdd2]

אל תמחק את מחיצות ווינדוס הללו בזמן התקנת לינוקס אם אתה מעוניין באיתחול כפול.
אם תמחק אותן לא יהיה לך עם מה ליישם איתחול כפול!
במידה ואינך בטוח, גבה את המידע שנמצא כל גבי מחיצות אלו לפני התקנת לינוקס.

“ntfs” “vfat” “fat” & “msdos” הינן מערכות קבצים שנמצאות בדרך כלל בשימוש תחת דוס ווינדוס NT/98/95/3.x.

כמבוא מהיר, לפניך סיכום זריז של סוגי מערכות הקבצים (“file systems”) בלינוקס מלווה בתיאור קצר.

העתקתי את המידע מתוך חוברת ההסברים של לינוקס (“linux manual pages”):

על ידי שימוש בפקודות `man fs` ו `man mount` (ביחד עם כמה הערות שהוספתי לאחר שהסתכלתי בקבצי קוד המקור ב `/usr/src/linux/fs`).

מערכות הקבצים שמודגשות הן אלו שסביר יותר שתשתמש בהן. מערכות הקבצים אחרות (אינן רשומות למטה) זמינות כתוספות (כמו מערכות קבצים דחוסות, מוצפנות וכו').

-minix

הינה מערכת קבצים אשר נמצאת בשימוש תחת מערכת ההפעלה Minix הראשונה שרצה תחת לינוקס. יש לה מספר מגבלות: גודל מחיצה של 64 מגה בלבד, שמות קבצים קצרים, חותמת זמן (time stamp), אחת וכו'. היא עדיין שימושית בפלופי וב RAM disks.

extended filesystem-ext

זוהי הארכה מפותחת של מערכת הקבצים מיניקס. היא דוכאה לחלוטין על ידי הגירסה השנייה של מערכת הקבצים המוארכת (ext2), ולבסוף תוסר מהקרנל.

ext2-היא מערכת קבצים בעלת ביצועים גבוהים שנמצאת בשימוש על ידי לינוקס לדיסקים קבועים ולמדיה שניתנת להסרה. מערכת הקבצים המורחבת השנייה עוצבה כהרחבה של מערכת הקבצים המוארכת (ext).

ext2 מציעה את הביצועים הגבוהים ביותר (במושגים של מהירות ושימוש במעבד) שמערכת קבצים שנתמכת תחת לינוקס יכולה להציע. בקיצור ext2 היא מערכת הקבצים העיקרית והדיפולטית של לינוקס.

ext3-הרכבה של מערכת הקבצים ext2 עם ג'ורנלינג (journaling). יש לה תאימות אחורה וקדימה עם ext2- ללא כל צורך בפירמוט מחדש או אובדן נתונים. (יש צורך אך ורק בהמרה מחדש של החלק המבוקש).

ext3-יכולה להיות מומרת בקלות ל ext2- גם כן ללא אובדן של נתונים או מידע. אני משתמש ב ext3 החל מאוקטובר 2001 והיא פשוטה ונטולת צרות. היא כלולה כאופציה בהתקנה מאז רד האט 7.2 ו מנדרייק 8.0. מומלץ מאוד שתשתמש במערכת קבצים זו.

xiafs-עוצבה והוטמעה על מנת להיות מערכת קבצים יציבה ובטוחה על ידי הרחבת הקוד של מערכת הקבצים מיניקס. היא מספקת את היכולות המבוקשות ביותר ללא מורכבויות מיותרות.
מערכת הקבצים xiafs כבר אינה מפותחת או מתוחזקת. משתמשים בה לעיתים נדירות.

msdos- זוהי מערכת קבצים שנמצאת בשימוש תחת דוס, ווינדוס וכמה מחשבי os/2. שמות הקבצים שלה לא יכולים להיות יותר גדולים מ-8 תווים עם תוספת אופציונלית של רווח 31 תווים.

umdos- היא מערכת קבצי דוס מורחבת שנמצאת בשימוש על ידי לינוקס. היא מוסיפה יכולות של שמות קבצים ארוכים, UID/GID, הרשאות POSIX, וקבצים מיוחדים (התקנים וכו') תחת מערכת הקבצים דוס, ללא הקרבת תאימות לדוס.

vfat-הרחבה של מערכת הקבצים דוס שבשימוש מייקרוסופט מאז ווינדוס 95 ווינדוס .NT
vfat מוסיפה יכולות לשמות קבצים ארוכים תחת מערכת הקבצים MSDOS.

proc-מערכת קבצים פסבדונית שנמצאת בשימוש כממשק למבני נתונים של הקרנל יותר מאשר קריאה ופיענוח של /dev/kmem.
הסתכל ב-man 5 proc.

iso9660- מערכת קבצים של CD-ROM שבנויה לפי הסטנדרט ISO 9660.
שתי הרחבות (רשומות למטה) נתמכות אוטומטית.
high sierra לינוקס תומכת במקדימה זו של הסטנדרט ISO 9660.
היא מוכרת באופן אוטומטי בתוך תמיכת לינוקס בסטנדרט הנ"ל.

rockridge
לינוקס תומכת גם בפרוטוקול השיתוף שמונחה על ידי rock ridge interchange protocol
הם משמשים על מנת לתאר את הקבצים במערכת ISO 9660 למארח יוניקס, ומספק מידע כמו שמות קבצים ארוכים, UID/GID, הרשאות POSIX, והתקנים. היא מזהה אוטומטית בתוך מערכת הקבצים ISO 9660, אשר נתמכת תחת לינוקס.

high performance filesystem-hpfs
בשימוש ב-os/2. מערכת קבצים זו נקראת בלבד תחת לינוקס עקב מחסור בתיעוד זמין.

sysv- הטמעה של System V/coherent filesystem ללינוקס. היא מיישמת את כל מערכות הקבצים של SystemV/386& Coherent, .xenix.

nfs-מערכת קבצים שמשמשת לגישה אל כוננים אשר ממוקמים על גבי מחשבים מרוחקים.

smb- מערכת קבצים של רשת שתומכת בפרוטוקול SMB.
פרוטוקול זה נמצא בשימוש על ידי קבוצות עבודה של מייקרוסופט ווינדוס ווינדוס NT, Lan Manager ו
שניתן למצוא ב: ftp://sunsite.unc.edu/pub/Linux/system/Filesystems/smbfs/.
הפקודה smbmount גם תספיק.

ncpfs- מערכת קבצים של רשת אשר תומכת בפרוטוקול NCP שנמצא בשימוש על ידי רשת Nvll NetWare.

--devpfs -- הינה מערכת קבצים פסבדונית, שבאופן מסורתי מוממרת ב/dev/pts.
בכדי להשיג פסוודו טרמינל, פרוסס (process) פותח את /dev/ptmx.
מספר הפסוודו טרמינלים נהיה זמין לפרוסס והסלייב (slave) פסוודו טרמינל הופך להיות
גיש כ <number>/dev/pts.

-fat אינה מערכת קבצים נפרדת, אלא חלק ממערכות הקבצים msdos, umsdos, & vfat.

-UFS מערכת קבצים שנמצאת בשימוש רחב המערכות הפעלה שונות.

swap-סוג מיוחד של מחיצה שמשמשת לגריפת נתונים מהזיכרון אל הדיסק הקשיח.

raiseffs-מערכת ג'ורנליזים חדשה לגמרי, זמינה כסטנדרט עם לינוקס מקרנל גירסה
2.4.1 ומעלה.
(ינואר 2001)

-hfs (=hierachical file system)
מערכת הקבצים של מקינטוש. זוהי גירסת בטא מאוחרת. לא ממולץ לעבודה עם נתונים
חשובים, אלא אם כן כקריאה בלבד.

ntfs מערכת הקבצים של מייקרוסופט ווינדוס-NT.
עדיין נסיונית תחת לינוקס, לא ממולץ למכונות יצרניות, אלא אם כן כקריאה בלבד.
(אוגוסט 2001)

1.10 המחיצה של מייקרוסופט ווינדוס תופסת לי את כל הדיסק הקשיח. האם אני יכול לכווץ/לחצות אותה בלי להתקין מחדש?

ייתכן. ישנו כלי שנקרא fipes.exe על גבי הדיסק של רד האט שיש לי שעושה בדיוק את זה.
בדוק את הסיפרייה \dosutils\fipsdocs\ בדיסק של רד האט שיש לך אם אתה מעוניין
בתיעוד.
אם אני הייתי במקומך, הייתי מגבה את כל המידע הקריטי שאני מחזיק לפני שהייתי עושה
משהו למחיצות שלי.
ישנם גם כלים מסחריים שמאפשרים לשנות את גודל המחיצה בלי לפגוע בתוכן שלה.
ההעדפה האישית שלי היא לבצע התקנה נקייה של ווינדוס על מחיצה אחת ייעודית.
אני משאיר קצת מקום לא מחולק למחיצות על הדיסק כדי שאוכל להשתמש בו מאוחר יותר
בשביל לינוקס. החלוקה ההוגנת של דיסק קשיח בין ווינדוס לבין לינוקס היא 50/50.
תוכניות של לינוקס נוטות להיות קטנות יותר אבל הן כוללות (כסטנדרט) רכיבים
שמייקרוסופט ווינדוס מציעה רק תמורת דולרים רבים שרתים (לא רק קליינטים) ל http, ftp,
telnet דואר כמה מסדי נתונים, שפות תיכנות, תוכניות עיבוד גרפי...

1.11 איך אני מתחיל את תהליך ההתקנה?

הכנס את דיסקט האיתחול אל תוך כונן הפלופי שלך, את התקליטור של רד האט אל תוך כונן
התקליטורים שלך, ואתחל את המחשב. אם אתה יכול לאתחל מכונן מהדיסק, הכנס את
התקליטור של רד האט אל כונן התקליטורים שלך ואתחל את המחשב.

תוכל להתחיל את ההתקנה גם מדוס (או תחת מצב דוס בווינדוס) בדיסק של רד האט שיש לי הרצתי את הפקודה ezstart.bat .

רוב ההתקנות מציעות לך לבחור בין מצב גרפי למצב טקסט. אתה צריך לבחור בטקסט אם המחשב שלך מוגבל בזיכרון.

1.12 האם ההתקנה של לינוקס קשה?

היא לא הייתה קשה בשבילי. נראה שלרוב המשתמשים החדשים, היא די עינינית ונטולת כאבים. זה גם תלוי בחומרה שלך ובבחירות ההתקנה, בדרך כלל היא נמשכת בין 40 דקות לשעתיים [פרט להתקנות ארוכות או בעייתיות עם מערכות ישנות בעלות זיכרון מוגבל מאוד-- לקח לי לילה שלם להתקין רד האט 6 על ה-33-486 מגה הרץ שלי עם 8 מגה זיכרון רם, המערכת עבדה ונראה שהיא לא עשתה כלום אבל בכל זאת ההתקנה הצליחה] שידרוגים מגירסאות קודמות אורכים יותר זמן ונוטים להיות יותר בעייתיים.

כמה משתמשים חדשים דיווחו שההתקנה הייתה סיוט טוטאלי בשבילם (בעיות חומרה? חוסר נסיון?) אם אתה נתקל בבעיות העצה שלי היא להתקין מערכת פשוטה, בלי להילחם על הרזולוציה הגבוהה ביותר לכרטיס המסך המפואר שלך או חומרה מתקדמת מאוד שיש לך. ניתן להוסיף/לקנפג הכל בשלב מאוחר יותר, אחרי שתהיה לך יותר הבנה לגבי איך הדברים עובדים על המערכת שלך. אפילו התקנה מחדש היא אופציה למשתמש חדש (מספרים שגורואים של לינוקס חושבים שזו בושה להתקין מחדש).

הרבה משתמשים חדשים נתקלים בבעיות כי הם קובעים רזולוציה גבוהה מדי (שאינה נתמכת או שנתמכת רק עם כיוון נוסף). שוב, זה לא יהיה חכם לשבור התקנה שלמה בשביל תמיכה בהתקן אחד ניתן לתמוך במכשיר ולכוון אותו מאוחר יותר.

1.13 אילו חבילות עלי להתקין?

תוכנה של לינוקס מגיעה בחבילות. לדוגמה, הדיסק של לינוקס מנדרייק שיש לי מכיל 1002 חבילות. מנדרייק 7.2 אורזת שני דיסקים של תוכנה--ההתקנה שלי של מנדרייק 7.2 הניחה 1123 חבילות על גבי הדיסק הקשיח שלי. רק שתדע, זו לא כל התוכנה שזמינה ללינוקס--אלא רק מבחר של חבילות שאנשי מנדרייק שמו בהפצה. מנדרייק נוטה לארוז יותר תוכנה מאשר רד האט.

לא משנה באיזו הפצה או גירסא, הדיסק מכיל חבילות שהופכות את מערכת ההפעלה הבסיסית (קרנל, ספריות, מבחר של כלים של שורת הפקודה לקינפוג ותחזוקה וכו') למבחר עשיר של קליינטי ושרתי רשת ביחד עם כלי הקינפוג והתחזוקה המתאימים, כמה אפליקציות במצב טקסט, מערכת X-WINDOS בסיסית, לפחות ממשק גרפי אחד (סביר להניח שכמה ממשקים), וכמות גדולה של אפליקציות בעלות ממשק גרפי.

תוכנית ההתקנה (מנדרייק או רד האט) תשאל אותך אילו חבילות להתקין. אם תבחר בהתקנת תחנת עבודה, אז החבילות שבדרך כלל נמצאות בשרתיים יישטו מההתקנה שלך (למשל, שרת הווב אפאצ'י) אם תבחר בהתקנת שרת, אז בדרך כלל אפליקציות שולחן העבודה לא יותקנו (כמו האפליקצייה הגרפית THE GIMP) תוכל גם להתקין את הכל, וזוהי הבחירה המועדפת עליי להתקנה ביתית.

לבסוף, תוכל לבחור את מבחר החבילות שאתה רוצה להתקין.
המשך לקרוא.

זה בהחלט רעיון גרוע לבחור התקנת חבילות/תוכניות לפי עד כמה השם שלהן נשמע לך מעניין-לכמה מהחבילות יש שמות לא רגילים בכלל ולעולם לא אצליח לנחש מה הן עושות. תוכל לגרום לנכות של המערכת שלך על ידי השמטת חבילות חיוניות (ספרייה). ייתכן גם שתתאכזב כשתתעקש להריץ אפליקציה עם שם מיוחד, שבמקרה כלולה בהפצה. באופן כללי, יכול להיות שתתעצבן מהפונקציונליות (או מאי הפונקציונליות) שהלינוקס ה"מותאמת" שלך מציגה. כמשתמש חדש, זה משתלם לעיתים לבטוח בבחירות הדיפולטיות שאורגנו על יד יותרי ההפצה שלך.

לכן, להתקנה הסופית שלי, הייתי נמנע מלהתפתות לבחור באפשרות בחירה ידנית של חבילות אלא אם כן אני מעוניין להתקין הכל. למתחילים, אני מעדיף את "ההתקנה הדיפולטית המקסימלית", או איך שהאופציה הזו לא נקראת בדיסק שלך.

1.14 איזה ממשק גרפי עליי להתקין KDE או GNOME?

במידה ויש לך מקום על הדיסק, את שניהם ללא ספק. מאוחר יותר תוכל להחליט אם אתה מעדיף את KDE, GNOME, או שולחן עבודה אחר, מה שלא תהיה הבחירה שלך, בטוח שתמצא את הספריות של שניהם מותקנות. ברגע שהספריות מותקנות, ניתן להריץ תוכניות של KDE תחת GNOME ולהיפך, שזה נהדר מכיוון שישנן אפליקציות נחמדות מאוד לשתיהן.

מבחינת שימוש בשטח דיסק, שולחן העבודה הוא רק חלק קטן מהמערכות KDE וGNOME ככה שאתה לא חוסך הרבה מקום על ידי השמטת שולחן העבודה מההתקנה ומתקין את הספריות בלבד. גם GNOME וגם KDE מגיעים עם מבחר נחמד של תוכניות וכלים, לכן בהחלט שווה להתקין את שניהם במלואם. לא שמעתי מעולם שהם עבדו בצורה שמפריעה או סותרת אחד את השני. לשימוש יומי, אני משתמש בשולחן העבודה של KDE מכיוון שהוא מרגיש יותר מוצק מ-GNOME. אם אתה רוצה ממשק יותר COOL בחר ב-GNOME אבל אל תתלונן אם הדברים לא עובדים כל כך טוב.

בנוסף הייתי מתקין גם "מנהלי חלונות" נוספים. הם בקושי תופסים מקום על הדיסק (חלק ממש זעירים) ועדיין הם יכולים להיות שימושיים בנסיבות מסויימות. אתה יכול להריץ כל אפליקצייה של KDE או של GNOME תחת כל אחד מהם, כל עוד הספריות של KDE ושל GNOME מותקנות.

KDE "רעב" יותר למשאבים.

על גבי חומרה ישנה (פנטיום 133 MHz) אני מעדיף את GNOME על פני KDE. מנהלי חלונות אחרים קלים יותר מ-KDE או מ-GNOME.

לכן, על גבי חומרה צנועה באמת הייתי בוחר במנהלי החלונות ה"אחרים".

1.15 סיימתי את ההתקנה. איך אני נכנס למערכת בפעם הראשונה?

כ root :

```
my_machine_name login: root
password: my password
```

בדוגמא למעלה, הקלדתי את המילה root בבקשת ה login . לאחר מכן, הכנסתי את הסיסמא שבחרתי במהלך ההתקנה. הסיסמא לא הופיעה על המסך כשהקלדתי אותה (אבטחה) . לאחר שנכנסתי למערכת, אני מוצא את עצמי במסוף טקסט.

אם התקנת את מסך הכניסה הגרפי, תהליך ה login נראה דומה, אבל מתרחש על גבי מסך x-windows (אם אתה נתקל כאן בבעיות הקלדה, זכור שסמן העכבר צריך להיות מעל קופסת הדיאלוג. מסך ה login x מיישם מדיניות של "פוקוס הולך בעקבות העכבר"). לאחר כניסה מוצלחת למערכת, הממשק הגרפי הדיפולטי שלי משוגר.

root הוא חשבון מיוחד עם כח אבסולוטי לגבי המערכת, והוא משמש לניהולה. כדאי שתיצור חשבון נוסף מאוחר יותר (או במהלך ההתקנה) לעבודה רגילה. המשך לקרוא כדי ללמוד איך לעשות את זה.

1.16 איך אני גורם ללינוקס לקרוס?

כ root , אתה יכול לעשות ככל העולה על רוחך, וזה כולל קריסה מואצת של המערכת. נסה (אם אתה באמת רוצה לגרום למערכת לקרוס) את:

```
cp /dev/zero /dev/mem
```

כ root אתה אפילו יכול למחור את כל הקבצים במערכת שלך בפקודה אחת תמימה (אל תעשה את זה):

```
rm -fr /
```

זה לא אומר שקל להפיל את לינוקס , אלא שלמנהל המערכת (root) יש את השליטה המוחלטת על המערכת. אתה יכול לגרום למיקרוסופט ווינדוס להיות לא שמישה על ידי מחיקת קבצים מ c:\windows או מ c:\windows\system .

תחת לינוקס, חשבונות מנהל המערכת ושאר המשתמשים הינם נפרדים. משתמשים רגילים יכולים לגעת אך ורק בקבצים ששיכים להם, ובאופן דומה להריץ תוכניות רק ב"אזורים" שלהם, ולא משנה כמה מבוגגת או עויינת התוכנית תהיה. ההפרדה בין חשבונות מנהל מערכת (administratortor) ובין חשבונות רגילים מוסיפה למורכבות המערכת, אבל היא עושה את לינוקס למערכת מרובת משתמשים (multi user) אמיתית. להבדיל מהגישה הישנה של מייקרוסופט. בגירסאות האחרונות של ווינדוס, מייקרוסופט נעה אל לעבר גישה יוניקסית.

אני נזכר באימרה ישנה שאומרת: "אלו שלא מכירים יוניקס מיועדים להמציא אותה מחדש".
מסקנה: אל תשתמש בחשבון root לעבודה שיגרתית. הוסף חשבון משתמש רגיל כמשימה
הניהולית הראשונה שלך והשתמש בחשבון זה להתנסות שלך עם לינוקס.

הנה הדרך לבצע זאת (עם root):

```
adduser johnbrown  
passwd johnbrown  
[הזן סיסמא טובה וארוכה עם מספר באמצע]  
[הקלד שוב את אותה הסיסמא בלי טעויות]  
exit  
[בצע לוג אין עם johnbrown]
```

החשבון root נועד לניהול והקמה בלבד. כ root לא הייתי עושה דברים/מריץ תוכניות שאני
לא כל כך מבין איך הם עובדים ומה הם עושים. לפחות לא על מחשב שאני מייעד לעבודה
אמיתית. באמת.

נתיב הלמידה שלי היה כזה:

1. התקנת לינוקס.
 2. משחק עם המערכת, נסיונות (כ root ולא כ root, מקים דברים, מתנסה. אני רוצה
להיות מסוגל לחזות את ההתנהגות של המערכת שלי—רק אז אדע שאני מבין
אותה).
 3. מתקין מחדש.
 4. במידה ואין לי מספיק זמן (אף פעם אין לי) חוזר ל 2.
- ברגע שלינוקס מותקנת כראוי על חומרה טובה, לינוקס של שורת הפקודה יציבה בצורה
אגדית- היא יכולה לרוץ במשך חודשים ואפילו שנים.
כמשתמש חדש, אתה יכול להיות בטוח שהתנהגות מוזרה של המערכת נובעת
מהמעשים שלך כ root או מחומרה לא תקינה.

1.17 האם אני יכול להשתמש בממשק גרפי כל הזמן?

אני מאמין שזה אפשרי תחת לינוקס, אבל אני לא חושב שזה יהיה יעיל לניהול המערכת.
הממשק הגרפי תחת לינוקס הוא נחמד מאוד, אבל הוא מחוויר בהשוואה למגוון
האפשרויות הזמינות תחת שורת הפקודה.
אני משתמש בממשק גרפי בכל יום תחת לינוקס וגם תחת ווינדוס, ואני מוצא אותו נהדר
להפעלת תוכניות, ומשימות שיגרתיות אחרות. בכל זאת לניהול מערכת ואוטומציה,
הממשק הגרפי אינו גמיש. ממשק שורת הפקודה הוא ממשק עשיר יותר מבחינתי,

ובנוסף הוא גם נותן לי קצת הבנה לגבי העבודה של המערכת שלי.

שני ציטוטים כדי לפשט את הנקודה הזאת:

"השימוש בממשק גרפי הוא די קל כי הוא מגביל אותך לכמה פעולות בסיסיות שאתה יכול ללמוד במהירות. אבל, אם הפעולות האלו הן לא מה שאתה צריך, אז יכול להיות שתתקל בבעיות".

" מאפיין אחד של מערכת ידידותית למשתמש הוא שהיא עושה מה שהמשתמש רוצה. במילים אחרות, המערכת הידידותית ביותר אינה בהכרח הפשוטה ביותר".

אני מוצא ששילוב של ממשק גרפי ושורת פקודה הוא מושלם לכיסוי כל הצרכים שלי. אם אתה נחוש בדעתך להשתמש בממשק גרפי בלבד ואינך מוכן ללמוד אפשרויות של שורת פקודה, שקול דחייה של התקנת לינוקס עד שכלי הממשק הגרפי יהיו מפותחים יותר, אלא אם כן יש לך מישהו שיעזור לך לנהל את המחשב שלך בינתיים.

מצד שני, אם יש לך חבר או חברה שלא כל כך מבינים שמחשבים, וכל מה שהיא/הוא עושים זה להפעיל חצי צריסר תוכניות שונות, אתה יכול להקים בשבילם ממשק גרפי נחמד עם אייקונים או כפתורים ותפריטים שהיא/הוא דורשים.

מנקודת מבט של המדריך הזה, שורת הפקודה פשוטה הרבה יותר לתיעוד. אייקונים ותפריטים נועדו להיות מותאימים ולכן האייקונים והתפריטים שלך יכולים להיות שונים משלי. בנוסף, כדי להקים אייקון או פקודה, אני צריך לדעת את הפקודה שעומדת מאחורי האייקון. בקצרה, הבנת שורת הפקודה הכרחית לכל סוג של עבודה מעבר לעבודה טריוויאלית. לא כללנו שום צילומי מסך במדריך הזה-אנחנו מוצאים אותם חסרי שימוש בלימוד על מחשבים.

1.18 איך לשדרג את הפצת הלינוקס שלי?

לשדרוג מערכת מלא (שידרוג דיסק הפצה $RH 7.0 < RH 7.2$) "לך בעקבות העיקרון המכובד של שידרוג מערכות יוניקס: התקן התקנה נקייה ורעננה והחזר בחזרה את הנתונים שלך. כן, אנחנו מדברים על פירמוט מחדש של המחיצות שלך והתקנה מאפס."

חלוקה למחיצות של הדיסק שלך בצורה נכונה יפשט שידרוגים עתידיים (קבצי הנתונים שלך יכולים לנישמר בספיריית ה /home).

רשימה שהכנתי לעצמי לשידרוג:

- כניסה למערכת כ root .

- יצירת עותק של הסיפרייה /etc במקום אחר שבו היא לא תושמד. (כונן זיפ, תקליטור). זה ייתן לי אפשרות להסתכל בתצורה הקודמת שלי למקרה שאני נתקל בבעיות.

- יצירת העתק של כל המידע החשוב בספרייה /home . זה רק למקרה שמשוה באמת לא יילך כמו שצריך, למשל, אם אני אפרמט בטעות את כל המחיצה.

- קביעה ורישום של כל נקודות ההמרה של מערכת הקבצים על ידי שימוש בפקודה `df` (`df`), במערכת שלי לדוגמא, היא מראה שמחיצת הבית שלי מומרת על גבי מחיצת דיסק קשיח נפרדת שנקראת `/dev/hda9`. ושהספרייה `/usr/local` נמצאת ב-`/dev/hsa8`.

- ביצוע התקנת לינוקס תוך שמירה ואי פירמוט של המחיצות `/home` ו `/usr/local`.
זה קריטי אם אתה מעוניין לשמור מידע.

- איתחול של ההתקנה החדשה וביצוע בדיקה שהיא אכן עובדת.

- יצירת זהות לכל משתמש במערכת עם השם ומספר הזהות הישן שלו: לדוגמא:

```
ls /home/maria/* -l
```

(בדוק את זהות המשתמש לחשבון "maria" ברשימת הקבצים במחיצת הבית שלה)

```
useradd maria -u 503
```

(צור חשבון משתמש בשם "maria" עם uid 503 – לדוגמא)

מכל ספריית בית של כל משתמש, מחק קבצים עם פוטנציאל לבעיות כמו:

"kde", ".ICE", וכו'. זו גם הזדמנות טובה להיפטר ולמחוק זבל. הקבצים יכולים להיות בעייתיים כי סביר להניח שגירסה חדשה של kde תתפקד טוב יותר עם ההגדות המעודכנות שלה (אני אצור אותן עם האיתחול הראשון).

```
cd /home/maria  
rm -fr .kde
```

וודא שכל קובץ בספריות הבית שייך למשתמש המתאים. למשל:

```
cd /home/maria  
chown -R maria *  
chgrp -R maria *
```

הבא: חלק 2, מקורות לקריאה, עזרה וכמה לינקים

חלק 2. מקורות לקריאה, עזרה וכמה לינקים

תוכן החלק הזה:

2.1 חומר קריאה לגבי לינוקס? [עמ' 36]

2.2 האם ישנה פקודת עזרה? [עמ' 37]

2.3 איזה שהוא מילון לגבי מונחים? [עמ' 38]

2.4 חיפוש ברשת [עמ' 38]

2.5 קבוצות דיון [עמ' 39]

2.6 לינקים ללינוקס באינטרנט? [עמ' 40]

2.7 קוד מקור--המקור האולטימטיבי [עמ' 43]

2.1 חומר קריאה לגבי לינוקס?

תקליטורי ההפצה של רדהאט מכילים המון תיעוד. חלק ממנו בפורמט html והשאר בפורמט טקסט פשוט. אתה יכול לקרוא את הכל תחת דוס או מייקרוסופט ווינדוס לפני שאתה מתקין את לינוקס.

לדוגמא, ניתן לצפות בהעתק של החוברת של רדהאט בכל דפדפן html תחת ווינדוס, נטסקייפ או אינטרנט אקספלורר. פשוט גש אל הקובץ:
D:\doc\rhmanual\manual\index.html (בהנחה שכונן התקליטורים שלך הוא D תחת ווינדוס).

בנוסף, בדוק גם את הספרייה \doc\LPD בשביל חוברות מצויינות של פרוייקט התיעוד של לינוקס. לדוגמא, אתה יכול לצפות במדריך ניהול הרשת של לינוקס על ידי גישה אל הקובץ \doc\LDP\sag\sag.html עם דפדפן האינטרנט החביב עלייך.

בנוסף, בדוק את הספרייה \doc\HOWTO בשביל מסמכי הHOWTO, הספרייה \doc\HOWTO\mini בשביל הMINIHOWTO ואת הספרייה \doc\FAQ בשביל סט של שאלות נפוצות בנושאים שונים ("FAQ=frequently asked questions"). לדוגמא, פקודות אלו ייתנו לך לקרוא את המסמך Linux-FAQ (מסמך טקסט פשוט) תחת דוס:

```
D:  
cd \doc\FAQ\txt  
edit Linux-FAQ
```

תחת לינוקס, אתה יכול לקרוא את אותו התיעוד מהתקליטור על ידי, לדוגמא, פקודה זו:

```
lynx /mnt/cdrom/doc/rhmanual/manual/index.html
```

זה יתחיל את lynx, דפדפן html פשוט שעובד במצב טקסט, לצפייה בחוברת של רדהאט. בבקשה שים לב, שתחת לינוקס, יש צורך להמיר (mount) תחילה את כונן התקליטור, ושהפקודה למעלה מניחה שנקודת ההמרה היא /mnt/cdrom. אתה יכול להשתמש גם בנטסקייפ ללינוקס, staroffice או בכל דפדפן html אחר כדי לצפות בחוברת של רדהאט ובתיעוד אחר בפורמט html. אתה יכול לקרוא מסמכי טקסט פשוטים מהתקליטור על ידי שימוש, לדוגמא, בפקודות אלו:

```
cd /mnt/cdrom/doc/FAQ/txt  
less Linux-FAQ
```

(הפקודה less מאפשרת לך לגלול את התוכן של קובץ טקסט.)

לאחר התקנת לינוקס, התיעוד, לא משנה איזה חלק התקנת, נמצא בספרייה /usr/doc או ב/usr/share/doc. אם לא התקנת את התיעוד, חשוב על התקנת כולו כעת, ייתכן וזה יהיה שווה את זה. לדוגמא, הספרייה, /usr/doc/LDP מכילה את חוברות פרוייקט התיעוד של לינוקס. פקודות אלו יאפשרו לך לצפות במדריך הניהול של לינוקס:

```
cd /usr/doc/LDP/sag  
lynx sag.html
```

בנוסף בדוק את `/usr/doc/HOWTO` בשביל מסמכי ה-HOWTO, ואת `/usr/doc/HOWTO/mini` בשביל ה-MINIHOWTO. המיקום של התיעוד נמצא לפעמים ב-`/usr/share/doc`.

לתיעוד נוסף ועדכני, הסתכל ב: <http://www.ibiblio.org/mdw/index.html>

2.2 האם ישנה פקודת עזרה?

ניתן להריץ את רוב הפקודות בלינוקס עם האופציה "`--help`". לדוגמא, פקודה זו תתן לך עזרה לגבי הפקודה `cp` (copy):

```
cp -- help | less
```

מידע מקיף יותר נגיש משורת הפקודה על ידי שימוש במה שנקרא חוברת השימוש `man topic` (manual pages). לדוגמא:

```
man cp
```

תציג את עמוד ההסבר שבחוברת לגבי הפקודה `cp`. דפי ההסברים (manual pages) הם מערכת העזרה הסטנדרטית תחת לינוקס, ומכילה עושר רב של פרטים, מידע טכני מאוד, אבל בדך כלל דורשים קצת מאמץ כדי להיות מובנים אם הקורא הוא משתמש חדש.

הפקודה `man` משתמשת בכלי קטן בשם `less` שייאפשר לך לגלול את הטקסט. השתמש במקשי החיצים כדי לגלול, לחץ על "q" כדי לפרוש. למעשה, `less` יכול לעשות יותר מזה. לחץ על "h" לעזרה כשרתה מריץ את `less`, או למד יותר על `less` על ידי שימוש בפקודה

```
man less
```

ישנה גם פקודת המידע `info topic`. לדוגמא:

```
info cp
```

שתתן לך עזרה לפקודה "`cp`" (copy). לעיתים קרובות `info` מכילה מידע דומה ל-`man`, אבל עדכני יותר. לרוע המזל, כלי הניווט של `info` אינו כה אינטואיטיבי, אישית אני משתמש ב-`man` לעיתים תכופות יותר. ישנו גם `pinfo` (תחליף לממשק של `info`, אולי קל יותר לשימוש מאשר `info`).

אם אתה לא זוכר בדיוק את שם הפקודה שבה אתה צריך להשתמש, נסה את `apropos`. לדוגמא, כדי להניב רשימה של פקודות שיש להן איזה שהוא קשר אל "copy", אני מבצע את זה משורת הפקודה:

```
apropos copy
```

הפקודה `what` דומה ל-`apropos`, אבל מתאימה רק מילות מפתח, בעוד ש-`apropos` מחפש במסד הנתונים השלם (מילות מפתח והתיאור שלהן). כתוצאה מכך, `what` נוטה להניב פלט קצר יותר (אולי רלוונטי יותר).

בכמה תוכניות שמונעות על ידי תפריט, לדוגמא כשמקנפגים את שירותי המערכת שלך על ידי ntsysv (או setup או linuxconf), אתה יכול ללחוץ על f1 בשביל מידע לגבי מה עושה שירות מסויים.

הרשימה של פקודות bash מובנות ניתנת להצגה על ידי הקלדת help בשורת הפקודה. ולאחר מכן עזרה לכל פקודת bash מובנת ניתן להצגה על ידי, לדוגמא:

help cd

bash היא המעטפת (shell) הסטנדרטית של שורת הפקודה, המקבילה של לינוקס למעבד שורת הפקודה של דוס "COMMAND.COM".

סביבת KDE כוללת דפדפן עזרה ("help browser") מבוסס ממשק גרפי, שניתן להפעלה על ידי האייקון המתאים ב"Kpanel" (תפריט המערכת, בדרך כלל בתחתית המסך). דפדפן זה יכול לשמש לגישה אל עזרה ספציפית לגבי KDE כמו גם אל דפי המידע של המערכת (manual pages). שולחן העבודה Gnome מכיל מערכת עזרה דומה.

אם אתה רוצה ללמוד על החבילות הרבות שמגיעות על התקליטור בפורמט rpm, יכול להיות שתצצה להשתמש בkpackagege מבוסס הממשק הגרפי (הקלד kpackagege בטרמינל X) כדי לדפדף בחבילות, להציג את המידע שהן מכילות, ולהתקין אותן אם אתה רוצה (ההתקנה חייבת להתבצע על ידי root). במקום kpackage, הפצות ישנות יותר משתמשות ב glint (רדהאט 5.2) או ב gnomrpm (רדהאט 6.0), שתי האחרונות איטיות יותר ופחות נוחות.

2.3 איזה שהוא מילון לגבי מונחים?

מילון זה די מקסימליסטי: "the new hackers dictionary" ידוע גם בשם "jargon file".
<http://www.tuxedo.org/~esr/jargon/jargon.html>
זהו לא רק מקור מצוין, אלא גם מספק קריא מבדרת מאוד. מומלץ.

כדי להוסיף בידור לבידור, הנה לינק נוסף שאני אוהב: "a girls guide to geek guys" (אם הלינק שבור-!-google it.א.א.)
<http://college.antioch.edu/totally/geek.html>
אם אתה זו את, יכול להיות שתעדיפי את: "a guy's guide to geek girls" (אם הלינק שבור-כנ"ל-א.א.)
www.eecis.udel.edu/masterma/guidetogeekgirls.html

רשימה די ארוכה של קיצורים שקשורים למחשבים וראשי תיבות ניתן למצוא ב:
http://www.geocities.com/ikind_babel/babel/babel.html

2.4 חיפוש ברשת

כרגע, מנוע החיפוש הטוב ביותר ברשת הוא Google, מדהים מה אתה יכול למצוא בעזרתו. גוגל ממש מהיר, מכיון שהוא רץ על לינוקס, אני לא צוחק. נסה את <http://www.google.com>. כדי לבחון אותו, בצע גלישת אגו (הקלד בתיבת החיפוש את: שם המשפחה שלך ומילה לבחירתך). ניתן להשתמש בגוגל כדי למצוא כמעט כל דבר רלוונטי לגבי לינוקס (או לגבי כל דבר אחר) ברשת. פשוט הקלד כמה מילות מפתח כדי למצוא את דוקומנטציית הלינוקס שאתה צריך.

2.5 קבוצות דיון

זה יכול להיות מקום קצת מפחיד לפעמים—נדמה שכל האנשים הכי מוזרים בעולם תמיד נוכחים בקבוצות הדיון. אני בדרך כלל בוחר להתעלם מהתגובות המטופשות או המעליבות או מהמיילים. לגבי אלו בעלי כוונות הזדון, אני מודיע למנהל הרשת של ספק הדואר שלהם. פרסומות שאני מקבל לאחר פרסום הודעה בקבוצת דיון נמחקות לפני שאני קורא אותן—אני יודע שאני לא היחיד שנוהג כך.

למרות החסרונות קבוצות דיון יכולות להיות דרך יעילה במיוחד למציאת המידע שאותו אתה צריך.

לפני שאתה פונה אל קבוצות דיון, הייתי ממליץ בחום לפנות אל ארכיבי הניוז של גוגל (<http://groups.google.com>), המוכרים כdejanews. זה מאגר ענק של קבוצות דיון ואתה יכול לחפש בתוכו בעזרת כלי חיפוש טובים. בדרך זו, אתה יכול לעיתים קרובות למצוא תשובה לשאלה שלך ללא הצורך לעבור דרך טונות של זבל, ובלי לחשוף את עצמך לכעס לאחר הצבת שאלה "שכבר נשאלה 10 פעמים השבוע". יכול להיות שתופתע מכמות המידע שזמין דרך הארכיב של גוגל.

ישנן כמה קבוצות דיון שמוקדשות ללינוקס ונראה שהן טובות יותר מקבוצות דיון אחרות (מנוהלות יותר טוב על ידי מומחי לינוקס?). לפניך רשימה קצרה:

news:comp.os.linux.announce (moderated)

(הפניות עוברות סינון על ידי moderator, שבודק אותן לפני הפרסום)

news:comp.os.linux.hardware
news:comp.os.linux.security
news:comp.os.linux.misc (miscellaneous)
news:comp.os.linux.advocacy

(השתמש בזו כדי לדון ביתרונות ובחסרונות של לינוקס ואולי גם להשוות בין לינוקס לבין מערכות הפעלה אחרות. זוהי קבוצת דיון מצויינת למי שאוהב להיכנס לוויכוחים שלא נגמרים)

news:alt.linux.sux

(כאן אתה יכול לכתוב/לקרוא את הדעות האמיתיות על לינוקס)

news:comp.os.linux.x (X-windows)
news:comp.os.unix

(קבוצת דיון ככלית לגבי יוניקס)

בבקשה שים לב שישנה אתיקה ("נטיקה") של קבוצות דיון, ואתה מסתכן בדחייה ואולי גם בבושה את עצמך ללהבות אם אתה בוחר להפר אותה. הנקודות העיקריות:

- אל תשלח נושא שאינו קשור לנושא של קבוצת הדיון.
- אל תשלח את אותה ההודעה לכמה קבוצות דיון באותו הזמן (cross-post).
- השתמש בascii נקי, אל תשלח אטצ'מנטס, תמונות, html וכו'.
- אל תפרסם (במיוחד לא מוצרים מסחריים).
- אל תכתוב באותיות גדולות בלבד.
- השתמש בנושא פשוט לתיאור הנושא(סתם לכתוב "עזרה" לא יעזור) והסבר בקצרה את הבעיה שלך. כולל תיאור של מספר גירסה, וזהה את החומרה שלך (אל תכתוב מודם 56k - usr - רשום את מספר הדגם).

כדי לקרוא קבוצות דיון (נקראות גם usenet), עלייך לקנפג את הגישה שלך אל שרת קבוצות דיון -newsserver. הדרך הפשוטה ביותר תהיה לקנפג את הדפדפן שלך על ידי ציון שרת הניוז שלך (ספק האינטרנט שלך, isp, צריך לספק לך את שם השרת) ואז להוסיף את קבוצת הדיון המתאימה אל תיבות הדואר המקומיות שלך, אם אינך יודע את שם שרת הניוז, נסה את: news.my_isp_provider_name.and_domain או אולי רק את "my_isp_provider_name.and_domain".

לקריאת ניוז, אני מעדיף את knod על פני נטסקייפ. התקנה ולמידה של קבוצות דיון היה בהחלט משהו ששוה את המאמץ.

2.6 לינקים ללינוקס באינטרנט?

יש בוודאי אלפי אתרים שמוקדשים ללינוקס. לפניך כמה לינקים ללינוקס שאני אוהב, לא מסודרים בסדר מסוים. אם אתה זקוק למשהו אחר, בטח תמצא הפניה מתאימה באחד מן האתרים האלו.

-<http://sunsite.dk/linux-newbie/>

האתר הראשי של מסמך זה (lnag) הוסף למועדפים.

-<http://www.linuxdoc.org/docs.html>

פרוייקט התייעוד של לינוקס- הבית של הרבה FAQs, howtos, minihowtos ומדריכים. תמיד מעודכן.

-<http://www.kalug.lug.net/linux-admin-FAQ>

FAQ לאדמניסטרטור לינוקס (שאינו ניוביי)

-<http://members.aanet.net/~swear/pedia/learning-linux.html>

האנציקלופדיה של גרי- ללמוד לינוקס. הוסף למועדפים.

-<http://jgo.local.net/linuxguide/>

עמוד הבית של גוש. מקור טוב ללימוד לינוקס.

-<http://www.control-escape.com/>

אתר טוב לניובייז.

-<http://www.linuxninja.com/linux-admin/>

ניהול לינוקס בקלות. מומלץ.

-<http://metalab.unc.edu/mdw/index.html#guide>

המון תיעוד על לינוקס. הוסף למועדפים.

-<http://www.frankelinux.com>

עוד אתר שעוזר לנייבייז.

-<http://www.easyfeed.com/~jgo/linuxguide>

מדריך נוסף לנייבייז.

-<http://www.slashdot.org/>

דיונים לחנונים, האקרים, גורואים וכו' (!.=)

-<http://www.freashmeat.org/>

עידכון לגבי תוכנות לינוקס שיוצאות עכשיו.

-<http://linuextoday.com/>

חדשות לינוקס- טוב לקריאה יומיומית. הוסף למועדפים.

-<http://www.llp.fu-berlin.de>

"פרוייקט המעבדה של לינוקס". למי שמתעסק במדעים.

-<http://linuxberg.com/>

פורטל גדול. יש להם הכל שם. אני אוהב את הדירוג שלהם לתוכנה של לינוקס, אני מתקין רק תוכנה שקיבלה שם 5 פינגווינים. מועדפים.

-http://counter.li.org/linuxcounter_eng.html

המונה של לינוקס. רשום את עצמך כלינוקסאי.

-<http://www.ap.univie.ac.at/users/havlik/album/linux-counter/>

ראה את המפות המרשימות של דניס הבליקס על הגדילה של לינוקס ועל התפוצה הגאוגרפית.

-<http://www.cl.cam.ac.uk/users/iwj10/linux-faq/index.html>

לינוקס faq

-<ftp://sunsite.unc.edu/pub/linux>

טונות של תוכנות ללינוקס בארכיב של sunsite. העדף את האתר.

-<http://stommel.tamu.edu/~baum/linuxlist/linuslist/linuslist.html>

אפליקציות ללינוקס

-<http://www.boutell.com/lsm/>

אפליקציות ללינוקס

-<http://www.linuxlinks.com/software/>

אפליקציות ללינוקס

-http://directorysearch.mozilla.org/computers/operaring_systems/linux

פורטל גדול (טוב יותר משל yahoo) עם לינקים מעולים לניובייז.

-<http://dir.yahoo.com/.../unix/linux>

הרשומות של yahoo לגבי לינוקס.

-<http://www.debian.org/>

האתר של דביאן לינוקס.

-<http://hardware.redhat.com/hcl/genpage2.cgi>

רשימת תאימות החומרה ללינוקס.

-<http://metalab.unc.edu.mdw/links.html>

הרבה לינקים מועילים שקשורים ללינוקס.

-<http://www.cse.unsw.edu.au/~conradp/linux/>

לינקים מעולים.

-<http://www.linuxlinks.com/>

ועוד לינקים.

-<http://www.gnu.org/>

האתר הראשי של gnu.

-<http://www.redhat.com/>

האתר של רד האט.

-<http://www.cs.helsinki.fi/u/torvalds/>

עמוד הבית של לינוס טרבולדה.

2.7 קוד מקור--המקור האולטימטיבי

מקור העיון האולטימטיבי תחת לינוקס הוא קוד המקור. אם התקנת אותו (מגיע עם הפצות סטנדרטיות, אנחנו באמת ממליצים להתקין אותו אם יש לך מקום על הדיסק), הוא נמצא ב-`/usr/src/linux` (the kernel source) וב-`/usr/src/RPM/sources` (קוד המקור לאיזון חבילות rpm).
כיצד יכול קוד המקור לשמש משתמש חדש? ובכן, הוא מכיל את כל ההערות והתיעוד עד לפרט הקטן ביותר. לדוגמא, בהמשך המדריך אנחנו מראים איך לקרוא/להגדיר חלק מהפרמטרים של סביבת הריצה של הקרנל דרך מערכת הקבצים `/proc`. אתה יכול לקרוא את התיעוד המלא לכל הפרמטרים האפשריים על ידי:

```
less /usr/src/linux/documentaion/proc.txt
```

כדי להתקין את המקור של הקרנל, הייתי בוחר את חבילות המקור המתאימות בזמן ההתקנה הראשית שלי. כדי להתקין מקור לחבילות אחרות שמגיעות עם ההפצה שלי, הייתי מכניס את ה"source CD" לתוך כונן התקליטורים ועושה משהו כמו (עם `cd` של רדהאט):

[מתקין את קוד המקור של גליון הנתונים gnumeric מהתליטור אל הדיסק הקשיח]

```
su  
cd /mnt/cdrom/SRPMS/  
rpm -ivh gnume<tab>
```

[עושה unzip לקוד המקור שזה עתה התקנתי]

```
cd /usr/src/RPM/SOUR<tab>  
tar -xvzf gnumer<tab>
```

[קורא את הקוד לפונקציות סטטיסטיות בgnumeric]

```
cd gnumeric/src/functions  
less fn-stat.c
```

זהו באמת המקור האמיתי לגבי איך גליון נתונים מסויים עובד, בלי צחוק.

עבור לחלק 3: פעולות בסיסיות

3.1 יסודות

3.1.1 שמות קבצים. [עמ' 46]

3.1.2 למה משמשות המחיצות השונות? [עמ' 49]

3.1.3 איך אני מריץ תוכנית? [עמ' 52]

3.1.4 כיצד אני יכול לשנות את התיב? [עמ' 54]

3.1.5 איך אני יכול לכבות את המחשב? [עמ' 55]

3.1.6 איך אני מתמודד עם תוכנה תקועה? [עמ' 57]

3.1.7 אפשרויות פקודה. [עמ' 61]

3.2 משתמשים, סיסמאות, הרשאות קבצים, ואבטחה

3.2.1 ספריות בית, root הוספת משתמש [עמ' 62]

3.2.2 קצת על אבטחת סיסמא [עמ' 63]

3.2.3 שכחתי את הסיסמא של root [עמ' 65]

3.2.4 שכחתי את סיסמת המשתמש שלי [עמ' 68]

3.2.5 ביטול או הסרה של חשבון משתמש [עמ' 68]

3.2.6 יש לי בעייה עם הרשאות קבצים, הרשאות קבצים ובעלות על קבצים. איך זה עובד?
[עמ' 69]

3.2.7 נגן mp3 שלי נחנק. הצליל לא נשמע טוב (איך להגדיר suid) [עמ' 73]

3.3 תזמון משימות עם "&", "at", "batch", ו-"cron"

3.3.1 איך אני מבצע פקודה ברקע? [עמ' 75]

3.3.2 איך אני מבצע פקודה בזמן מסויים (על ידי שימוש ב"at" או "batch")? [עמ' 75]

3.3.3 איך אני מגדיר את cron? [עמ' 77]

shell 3.4

- 3.4.1 מה זה shell (מעטפת) והאם אני רוצה להשתמש באחרת? [עמ' 79]
- 3.4.2 כיצד אני מבצע התאמה אישית למעטפת שלי? [עמ' 80]
- 3.4.3 צבע בטרמינל טקסט [עמ' 81]
- 3.4.4 כיצד אני מדפיס סימנים בקונסול, באפליקציות מצב טקסט, וב-X? [עמ' 82]
- 3.4.5 איך אני כותב סקריפט מעטפת פשוט? [עמ' 83]
- 3.4.6 המשמעות של ציטוטים [עמ' 85]
- 3.4.7 הפניה מחודשת של פלט/קלט [עמ' 85]
- 3.4.8, תוי מעטפת מיוחדים (metachacters) [עמ' 88]

3.5 התקנת חבילה ומנהל החבילות rpm?

- 3.5.1 איך אני מתקין תוכנית שהורדתי מאינטרנט? [עמ' 90]

3.1 יסודות

3.1.1 שמות קבצים

לינוקס היא case sensitive . למשל, myfile, MyFILE, & myFILE הם שלושה קבצים שונים.
גם הסיסמא שלך והשם שלך, איתו אתה נכנס למערכת, הינם case sensitive .
(זה מסורתי, גם יוניקס וגם שפת התיכנות C הם case sensitive) . מוסכמת שמות לקבצים ולספריות הינן זהות. כל הקבצים והספריות שאני יוצר (לעצמי כמשתמש) הם באותיות קטנות, אלא אם כן ישנה סיבה מיוחדת לשנות את זה . רוב הפקודות של לינוקס גם הן נכתבות באותיות קטנות.

שמות קבצים תחת לינוקס יכולים להכיל עד 256 תווים ובדרך כלל מכילים אותיות, מספרים, נקודות ".", קווים תחתונים "_" ומקפים "-".
לא מומלץ להשתמש בתווים מיוחדים: כוכבית "*", סימן שאלה "?", רווח " ", סימן דולר "\$", ו"&", וכו'.
זה נובע מהעובדה שלתווים מיוחדים אלו יש משמעות מיוחדת לshell של לינוקס. (shell – מעטפת- זה משהו כמו command.com תחת ווינדוס). שם קובץ יכול להכיל רווח, אבל לא היינו ממליצים על זה—אנחנו משתמשים בקו תחתון "_" במקום.

לא ניתן להשתמש בקו נטוי "/" כחלק משם קובץ מכיוון שהוא מייצג את ראש עץ הסיפרייה, ומשמש כמפריד בנתיבי שמות (מקביל ל "\" בדוס).

כמו בדוס, אי אפשר לקרוא לקובץ ". או ".." (נקודה או שתי נקודות)—כי הן מסמלות את הסיפרייה הנוכחית ואת הסיפרייה האם בהתאמה.
ממש כמו בדוס.

לפנייך המשמעויות של כמה תווים מיוחדים (או metacharacters):

*= מתאים לכל רצף של אפס תווים או יותר, פרט ל "." (נקודה) בהתחלת קובץ.

?=מתאים לכל תו יחיד.

[abC1]=מתאים לתו יחיד בסט . בדוגמא זו הסט מכיל: ; a, b, C ו-1.

[a-z]=מתאים לכל אות קטנה.

[A-F]= מתאים לכל אות גדולה מ A ל F .

[0-9]- מתאים לכל סיפרה אחת.

[a-zA-Z0-9]=מתאים לכל אות (קטנה או גדולה) ולכל סיפרה.

גם התו "\" back slash מיוחד. הוא גורם לתו מיוחד להיות בעל משמעות (המשך לקרוא).

דוגמאות. הפקודה הזו תציג כל קובץ בסיפרייה הנוכחית, פרט לקבצים שמתחילים ב". (נקודה):

ls *

מקבילה לפקודה זו תהיה הקלדה של ls בלבד או dir בלי ה *. קבצים עם שמות שמתחילים בנקודה ". לא יוצגו מכיוון האות הראשונה בשם קובץ לא תואמת ל" ". תחשוב על קבצים שמתחילים בנקודה כמקבילה של קבצים נסתרים בדוס. השתמש ב ls-a (מציג רשימה עם האופציה "all") או ls.* כדי להציג קבצי נקודה. קבצי נקודה שכיחים ב ספריות user home ובדרך כלל משמשות כדי להכיל קונפיגורציות ברמת משתמש.

פקודה זו תציג כל קובץ (בסיפרייה הנוכחית) שמכיל נקודה:

ls *.*

פקודה זו תציג כל קובץ שמכיל שתי נקודות:

ls *.*.*

שים לב בבקשה שללינוקס אין "הארכות שמות קבצים" כמו שיש לדוס, אבל עדיין תוכל להשתמש בהן. למשל, יכול להיות שיהיה לי קובץ בשם my_text.txt.zip. שמות קבצים אחרים (כמו "micro~1.doc" למשל) של דוס נעדרים כאן.

פקודה זו תאתר (בכל מערכת הקבצים) כל קובץ עם התוספת "htm" כשאחרי תו כלשהו:

locate *.htm?

פקודה זו תראה את כל הקבצים בסיפרייה הנוכחית, שמתחילים ב "a" או "b", או בכל אות גדולה:

ls [abA-Z]

פקודה זו תציג כל קובץ שמתחיל ב "a" ונגמר ב "n" :

ls a*n

השלמה אוטומטית בשורת הפקודה. (autocomplation) זוהי אופציה נהדרת של שורת הפקודה—אני משתמש במקש [tab] המון כדי לחסוך הקלדה מיותרת. זה הופך את כל ההתעסקות עם שמות קבצים ארוכים לקלה ומהירה. לדוגמא, שימוש בשם קובץ כזה בשורת הפקודה ממש לא מהווה בעייה, במידה ואתה משתמש בהשלמה אוטומטית:

dir Eurosong\ 2000\ olson\ brothers\ -\ fly\ on\ the\ wings\ of\ love\ \ (denmark\)/mp3

אני פשוט מקליד:

```
dir Eu<tab>
```

ואם אין קבצים אחרים שמתחילים ב "Eu", שאר הקובץ מוקלד אוטומטית בשבילי. אחרת, אני אצטרך להסתכל באפשרויות (שמודפסות בשבילי) ולהקליד עוד תו אחד או שניים כדי להבדיל בין שם הקובץ המבוקש לבין השאר. ה backslash "\" בדוגמא למעלה הוא חלק משם הקובץ.

צרות עם שמות קבצים מוזרים. רוב הבעיות האלו יכולות להיפתר על ידי שימוש בהשלמה אוטומטית. בנוסף כדי ל"נווט" עם שמות שמכילים סימנים מיוחדים, אני משתמש בזוג גרשיים ' ', כך שהסימנים המיוחדים מצוטטים והמעטפת (shell) לא מפרשת אותם כמשמעותם. כדוגמא, כדי לשנות את שם הקובץ "my file*" אני מקליד:

```
mv 'my file*' filename_without_weird+characters.txt"
```

שים לב שהשתמשי בזוג ' מרכאות לציטוט. ציטוט עם זוג " (גרשיים) הינו באופן כללי חלש יותר מאשר ציטוט עם ' . אם תשתמש בגרשיים " , סימנים מיוחדים אחדים יכולים להתפרש על ידי המעטפת.

בהתאם למסורת של יוניקס, בלינוקס, אתה יכול ליצור קבצים בעלי שמות שמכילים כמעט כל תו או סימן, כולל תוים לא מודפסים (control). תוים אלו אינם נפוצים, אבל אם אתה נתקל בקובץ כזה, הוא יכול לגרום לך להרגיש ממש משונה. הייתי משנה שם קובץ כזה על ידי שימוש סימן מיוחד שמוקם בזהירות בשם הקובץ. קודם כל הייתי משתמש ב ls כדי לראות אם הפעולה שלי אכן הצליחה ואחר כך משנה את שם הקובץ:

```
ls -l myfile*y.html
```

```
mv myfil*y.html myfile.html
```

כדוגמא לבעייה אולי הכי מוזרה שאתה עלול לפגוש בשימוש בתוים לא ממוצאים בתוך שם קובץ, נסה ליצור קובץ עם שם שמתחיל במקף ואז הסר אותו—נדמה שאין דרך לעשות את זה (מכיוון שמקף מסמן אופציות לפקודה) הפקודה :

```
dir > -junk
```

תיצור קובץ מצחיק (כמו בדוס, הסימן > מפנה את הפלט מהפקודה dir אל קובץ בשם -junk). מכיוון שהדרך הרגילה להסרת הקובץ -junk לא עובדת, אני משתמש ב:

```
rm ./-junk
```

ההתחלה עם נקודה קו נטוי (dot slash) מתפרשת כ"בספרייה הנוכחית" ומשרתת כאן את הרצון להסתיר את המקף כדי שלא ייתפרש כאופצייה לפקודה chmod. הנקודה כאן היא שעדיף שאדבוק עם מוסכמות שמות מסורתיות מאשר אתקל בסיבוכים מיותרים.

מלבד שימוש בהשלמה אוטומטית, מרכאות וגרשיים, אני יכול לעשות מניפולציה על קבצים על ידי שימוש ב קו נטוי אחורי (backslash).
לדוגמא, אני יכול ליצור קובץ בעל השם המשונה *? [על ידי שימוש בפקודה :

```
touch \*\?*
```

(הפקודה touch יוצרת קובץ ריק או, במידה והקובץ קיים, מעדכנת את התאריך/זמן של העידכון האחרון).

3.1.2 למה משמשות הספריות השונות?

בקצרה, לינוקס מכילה חמש מערכות קבצים. מערכות קבצים אלו שוכנות על גבי דיסק פיזי אחד או יותר (ואחר) או על גבי מחיצה אחת או יותר (ואחרת), תלוי בגודל ובצורך של המערכת שלך. (מערכת קבצים אחת יכולה להיות מופצת בין כמה התקנים פיזיים, במקרה הצורך).

the root "/" filesystem

מכילה את מערכת ההפעלה הבסיסית וכלי תחזוקה. התוכן של מערכת קבצים זו אמור להספיק להפעלה של המערכת ולביצוע תחזוקת חירום ותיקון אם הם יש צורך בכך.

/usr filesystem

מכילה את כל הפקודות, ספריות, תיעוד, וקבצים אחרים שלא משתנים בזמן תיפקוד נורמלי. סיפרייה זו מכילה גם אפליקציות רבות שמגיעות עם ההפצה שלך, netscape למשל.

/var filesystem

מכילה קבצים שמשתנים: ספריות spool, קבצי יומן (log), קבצי lock, קבצים זמניים, ודפי תיפעול (manual pages).

/proc filesystem

מכילה קבצים הזויים ודמיוניים בהחלט. הם לא קיימים באמת על גבי הדיסק והם אינם תופסים מקום (למרות ש ls -l תראה את הגודל שלהם) כשאתה צופה בהם, אתה למעשה ניגש למידע שמאוחסן בזיכרון. משמש לגישה למידע לגבי המערכת.

החלקים של מערכת הקבצים root הם:

/bin – executables (binaries) שבשימוש על ידי המערכת בזמן האיתחול, שיכולים להיות בשימוש על ידי משתמשים רגילים.

—/sbin executables (system binaries) לא מכוונים לשימוש על ידי משתמשים רגילים (משתמשים עדיין יכולים להשתמש בהם, אבל סיפרייה זו אינה הנתיב שלהם PATH).

—/etc קבצי קינפוג של כלל מערכת ההפעלה שלך.

`/root`—ספריית הבית של מנהל המערכת. (נקרא משתמש על – `super user` או `root`)

`/dev`—קבצי התקנים. התקנים מופיעים בלינוקס כהתקנים כך שחומרה הופכת אבסטרקטית וקל לכתוב אליהם ולקרוא מהם.

`/mnt`—נקודות המרה למדיה ניידת (פלופי, `cdrom`, כונני זיפ), מחיצות למערכות הפעלה אחרות (ווינדוס), שיתופי רשת, וכל דבר אחר שמומר אל מערכת הקבצים באופן זמני. באופן נורמלי מכילה תת ספרייה נפרדת לכל נקודת המרה. תכולת כוננים/שיתופים אלו מופיע בתת ספריות אלו—אין להם אותיות כונן (C או D) בלינוקס.

`/lib`—ספריות משותפות לתוכניות ששוכנות על גבי מערכת הקבצים `root` ועל מודולי קרנל.

`/boot`—קבצים שבשימוש טוען איתחול (LILLO או GRUB), מה שנטען ראשון כשהמחשב מאתחל ואולי גם מה שנותן לך את האפשרות לבחור מערכת הפעלה לאתחול, אם יש לך יותר ממערכת הפעלה אחת על המחשב. בדרך כלל מכיל את הקרנל של לינוקס (בקובץ הדרוס, `vmlinuz`), אבל יכול להיות מאוחסן גם במקום אחר, במידה ו `lilo` מקונפגת לדעת היכן.

`/opt`—אפליקציות אופציונליות גדולות, כמו `kde` תחת רדהאט 5.2 (תחת רדהאט 6 `kde` מופץ ככל תוכנית `x-windos`, `executables` עיקריים נמצאים ב `/usr/bin`).

`/tmp`—קבצים זמניים. ייתכן וספרייה זו תתנקה אוטומטית.

`/lost+found`—קבצים ששוקמו בזמן תיקון של מערכת הקבצים.

החלקים הכי מעניינים של `/usr filesystem` הם:

`/usr/x11r6`—מערכת ה `x-windos` (גירסה 11, הפצה 6).

`/usr/x11`—זהה ל `/usrx11r6` (לינק סימבולי).

`/usr/x11r6/bin`—הרבה אפליקציות של `x-windows` ששוכנות בתת הספריות שלהן במקום אחר.

`/usr/doc`—תיעוד של לינוקס(במערכות חדשות, זה זז ל `/usr/share/doc`).

`/usr/share`—מידע עצמאי מארכיטקטורת המחשב שלך.

`/usr/bin` ו-`/usr/sbin`—דומה למקבילות שלהן במערכת הקבצים `root` (`/bin` ו `/sbin`), אבל אינן דרושות לאיתחול בסיסי (בזמן תחזוקת חירום) רוב הפקודות יישכנו שם.

`/usr/local`—האפליקציות שמותקנות על ידי האדמיניסטרטור המקומי (ייתכן שכל אפליקציה בתת ספרייה נפרדת). לאחר ההתקנה ה"ראשית" ספרייה זו ריקה. התוכן שלה אמור לשרוד התקנה מחדש נורמלית או שידרוג של מערכת ההפעלה.

`/usr/local/bin`—אולי `executables` קטנים יותר של המשתמש, פלוס לינקים סימבוליים `executables` גדולים יותר שנמצאים בתת ספריות נפרדות תחת `/usr/loacl`.

חשוב להבין שכל הספריות מופיעות בעץ ספרייה יחיד, אפילו אם הספריות נמצאות על גבי מחיצות שונות, דיסק פיזי אחר, או אפילו אם הן נמצאות ברשת. לכן אין אותיות כונן כמו בדוס תחת לינוקס. מה שיהיה כונן תחת דוס יופיע כתת סיפרייה במיקום המרה מיוחד תחת לינוקס.

מערכת הספריות (מדורים) מיושמת היטב תחת רוב ההפצות של לינוקס. היא גם די דומה למערכת שנמצאת תחת מערכות יוניקס מסחריות רבות.

עוד קצת על מערכת הקבצים `/proc` (רק לסקרנים באמת).

מערכת הקבצים הפסוודונית `/proc` הינה מערכת קבצים ששוכנת בזמן אמיתי בזיכרון, ושעוקבת אחר המצב של ליבת מערכת ההפעלה (הקרנל—`kernal`) ואחרי התהליכים שרצים על המחשב שלך. מערכת הקבצים `/proc` היא ווירטואלית לחלוטין, היא אינה נכתבת על גבי דיסק כל שהוא או על מדיה אחרת, היא קיימת אך ורק בזיכרון המחשב, ומעודכנת באופן תמידי על מנת לשקף שינויים במערכת שלך. הגודל של מערכת הקבצים `/proc` הוא תמיד אפס וזמן העידכון האחרון הוא התאריך הנוכחי. במקרים מסויימים, ניתן לשנות את הגדרות המערכת על ידי שינוי ידני של תוכן הקבצים אשר בתוך `/proc`. כלים רבים של לינוקס משתמשים במערכת הקבצים `/proc` כמקור המידע שלהם, `dmesg`, `....ps`, `top`

התוכן של `/proc`.

ספריות עם שמות מספריים כמו "1" "170" "4980" הינם תגי זיהוי של תהליכים (`processes`) שרצים במחשב שלך. כל ספרייה מכילה כמה קבצים; `cmdline` (מכיל את כל שורת הפקודה ששימשה לעורר את התהליך), `cwd` (לינק סימבולי ל `cwd` של התהליך), `environ` (משתני הסביבה שמוגדרים על ידי תהליך מסויים באופן `VARIABLE=value`), `exe` (לינק סימבולי לקובץ `executable` שהתהליך הנוכחי מקושר אליו), `fd` (רשימה של מילות מפתח שנפתחו על ידי התהליך), `mpas` (משרשר—`pipe`— שיכול לשמש לגישה אל זיכרון התהליך) [`pipe=pipe` מורה למערכת ההפעלה להפנות פלט של תוכנה x לקלט של תוכנה y—א.א]

`root` (לינק סימבולי אל הספרייה שהיא מערכת הקבצים `root` לגבי תהליך מסויים), `stat` (מידע לגבי מצב של התהליך).

קבצים נוספים ב`/proc`:

`/proc/cpuinfo`—מידע לגבי המעבד, כמו הסוג שלו, דגם, יצרן, וביצועים.

`/proc/devices`—רשימה של מנהלי התקנים שמקונפגים אל תוך הקרנל שרץ כרגע.

`/proc/dma`—ערוצי DMA שנמצאים בשימוש כרגע.

—/proc/filesystems סוגי מערכות קבצים שמקונפגים אל תוך הקרנל.

/proc/interrupts -אותות פסיקה שנמתאים בשימוש, וכמה מכל אחד היו בשימוש.

—/proc/ioprocs פורטים של I/O שבשימוש כרגע.

לדוגמא, אני יכול לקרוא את מידע לגבי המעבד שעל המערכת על ידי הפקודה הבאה:

```
cat /proc/cpuinfo
```

3.1.3 איך אני מריץ תוכנית?

הקלדת השם של התוכנית (executable) בשורת הפקודה לא מניבה תוצאות? ישנן שלוש אפשרויות.

האפשרות הראשונה: לא הקלדת את שם התוכנית בצורה תקינה. בדוק איך כתבת אותה - לינוקס רגישה לנושא הזה. לדוגמא, הקלדת "pico" או "PICO" לא יפעיל את העורך pico .

האפשרות השנייה:אולי התוכנית אינה בנתיב (path) שלך. תחת לינוקס (או יוניקס), executable חייב להיות בנתיב שלך כדי להריץ אותו, והספרייה הנוכחית אינה הנתיב שלך.

הקלד את הנתיב המלא אל exec לפני שמו, או בצע:

```
cd the_program_directory  
./program_name
```

אתה חייב לשים נקודה וסלאש לפני שם התוכנית אחרת התוכנית לא תרוץ. (אי הקלדת הסיפרייה בנתיב היא תכונת אבטחה . היא מקשה קצת את החיים על "סוסים טרווינים". "סוס טרוויני" הוא למעשה תוכנה זדונית שמעמידה פני תוכנה אחרת ממה שהיא באמת.) הנקודה מתפרשת כ"ספרייה הנוכחית", והסלאש "/" מפריד בין שם הסיפרייה ושם הקובץ (בדיוק כמו בדוס).

אתה יכול לבדוק את הנתיב שלך על ידי:

```
echo $PATH
```

כדי ללמוד איך לשנות את הנתיב שלך, או כדי להוסיף את הספרייה הנוכחית שלך אליו, הסתכל בתשובה הבאה [3.1.4]

אם התוכנית שלך אבודה איפה שהוא בעץ הסיפרייה שלך, ייתכן ותרצה למצוא אותה על ידי (לדוגמא):

```
find / -name "netscpae"
```

כדי למצוא קובץ בשם "netscape" על ידי חיפוש מספריית השורש "/". תוכל להשיג להשיג את אותה תוצאה בצורה מהירה יותר על ידי שימוש ב:

```
locate netscape
```

(locate פועלת מהר יותר מכיוון שהיא נשענת על מסד נתונים על קבצים שמערכת שלך אשר נבנה מראש. מסד נתונים זה מעודכן על ידי הליך cron שבדרך כלל רץ בלילה, אז אל תסמוך על locate אם אתה נוהג לכבות את המחשב שלך בלילה, או אם אתה מחפש קובץ שזה עתה התקנת).
[ניתן לבנות את מסד הנתונים גם באופן ידני על ידי הקלדת הפקודה: updatedb:א.א].

שים לב בבקשה שהנתיב בדרך כלל שונה למשתמש root ולמשתמשים רגילים (הנתיב של root כולל את: /sbin ואת /usr/sbin בעוד שהנתיב של שאר המשתמשים לא).
לכן משתמשים לא יכולים להריץ פקודות שממקומות בספריות "/sbin" אלא אם הם מציינים את הנתיב המלא אל הפקודה. בנוסף אם אתה הופך להיות סופר יוזר (super user) על ידי הרצת הפקודה su, אתה יורש את הנתיב של המשתמש, וכדי לבצע את הפקודה שממוקמת ב /sbin, עלייך לציין את הנתיב המלא.

באופן הפוך, אם אתה צריך לדעת היכן ממקומת תוכנית שנמצאת בנתיב שלך על גבי המערכת (התוכנית רצה על ידי הקלדת שמה בכל מקום במערכת, אבל אתה מעוניין לדעת היכן היא נמצאת), אתה יכול להשתמש במשהו כמו זה:

```
which netscpae
```

which תראה לך את הנתיב המלא אל התוכנית שנקראת netscape .

האפשרות השלישית: אולי הקובץ אינו בר הרצה. אם הוא אמור להיות, שנה את ההרשאות כדי להפוך אותו לכזה. (כroots או כ משתמש שהוא בעל הקובץ):

```
chmod a+x my_file
```

פקודה זו תהפוך את הקובץ "my_file" לבר הרצה לכל המשתמשים. בדוק אם זה עובד על ידי:

```
ls -l my_file
```

שים לב בבקשה שתחת לינוקס (או יוניקס), אין זו הסיומת של הקובץ (exe או com או bat למשל) שהופכת את הקובץ לבר הרצה. הקובץ צריך להיות במצב גישה שמאפשר לו להיות כזה ("executable file access mode") שאינו דומה כלל לאופן בו עובדים האטריביוטים של קבצים תחת דוס.

3.1.4 כיצד אני יכול לשנות את הנתבי?

בדרך כלל, אינך צריך לשנות את הנתבי שלך, אבל זה מועיל מאוד להבין מה זה נתבי (PATH).

הנתבי הוא רשימה של ספריות שבהן נערך חיפוש כשאתה מבקש להריץ תוכנית. אתה יכול לבדוק את הנתבי שלך על ידי שימוש בפקודה זו:

```
echo $PATH
```

שבמערכת שלי, מראה שהנתבי של המשתמש yogin הוא:

```
/opt/kde/bin:/usr/local/bin:/bin:/usr/bin:/usr/X11R6/bin:/home/yogin/bin
```

הנקודותיים ":" משמשות כמפרידות לכן הנתבי מייצג רשימת ספריות כלהלן:

```
/opt/kde/bin  
/usr/local/bin  
/bin  
/usr/bin  
/usr/X11R6/bin  
/home/yogin/bin
```

הנה פלט מהפקודה "echo \$PATH" תחת המשתמש root במערכת שלי:

```
/opt/kde/bin:/sbin:/bin:/usr/sbin:/usr/bin:/usr/X11R6/bin:/root/bin
```

אתה יכול לשנות את הנתבי לכל המשתמשים במערכת על ידי עריכה של הקובץ `/etc/profile` והתאמת השורה שמתחילה ב "`PATH=`". אני עושה את זה על ידי שימוש בעורך `(root)` pico:

```
pico -w /etc/profile
```

(האופציה `-w` מכבה את `warp` לשורות ארוכות.)

היכנס מחדש למערכת כדי שהשינוי יוחל. כדי לקבוע את הנתבי למשתמש יחיד בלבד, ערוך את הקובץ `/home/user_login_name/.bahe_profile` (שים לב בבקשה לנקודה לפני שם הקובץ—קבצים שמתחילים בנקודה בדרך כלל בלתי נראים, אתה צריך להשתמש ב `ls -a` על מנת לראות אותם).

אם אתה באמת מעוניין שהספרייה הנוכחית תהיה בנתיב שלך, הוסף "." (נקודה) לנתיב שלך.
כששימוש במקום בו אמור להיות שם הספרייה, נקודה מתפרשת כ"ספרייה הנוכחית".
ההנחיה לנתיב ב `/etc/.bash_profile` יכולה להראות כך:

```
PATH="$PATH:$HOME/bin:."  
export PATH
```

פקודה זו לוקחת את התוכן של המשתנה הסביבתי שנקרא PATH (כפי שנקבע לכל המשתמשים ב `/etc/profile`), ומחיל עליו את השם של ספריית הבית שלך כפי שנקבע על ידי המשתנה HOME בתוספת `"/bin"` ואז נקודה.

לבסוף, הפקודה מחילה את חזרה את המשתנה שנקרא PATH. זה הכרחי להשתמש בפקודה `"export"` לאחר שינוי PATH או כל משתנה סביבה שקשור למשתמש, ככה שמשתנה הסביבה נראה מחוץ לסקריפט שקובע אותו.

3.1.5 איך אני יכול לכבות את המחשב שלי?

בטרמינל טקסט, לחץ על `<ctrl><alt>` (הצדעת "שלוש האצבעות", לוחצים על שלושת המקשים ביחד), חכה שתהליך הכיבוי ייסתיים, ואז כבה את המכונה שלך רק לאחר שהיא מאתחלת שוב.

אם אתה נמצא ב X-windows, קודם עבוד לטרמינל טקסט על ידי לחיצה על `<ctrl><alt><f1>` (ביחד). אל תכבה את המכונה שלך ללא הכיבוי הראוי מכיון שאתה עלול להיתקל בהודעות שגיאה לגבי הדיסק בפעם הבאה שתאתחל. (בדרך כלל, השגיאות שנובעות מכיבוי לא תקין ייתוקנו באופן אוטומטי בזמן האיתחול הבא, אבל לפעמים בעיות רציניות יותר יכולות לנבוע מכיבוי לא נכון, ואז תאלץ לתקן את הקבצים ידנית או להתקין מחדש!)

אם אתה מעדיף שהמחשב שלך יעצור (יכבה) לאחר שאתה לוחץ `<ctrl><alt>` (במקום האיתחול הדיפולטי), אתה יכול לקבוע את זה על ידי עריכת הקובץ `/etc/inittab`. הקובץ מציין משהו כמו זה:

```
#trap CTRL-ALT-DELETE  
ca::ctrlaltdel:/sbin/shutdown -t3 -r now
```

כ `root`, החלף את האופציה `"-r"` ל"`-h`".

השורה שמתחילה ב"`#`" היא רק הערה (לבני אנוש בלבד, אין לה כל השפעה על המחשב). האופציה `"-t3"` מורה לפקודת הכיבוי (`shutdown`) לחכות 3 שניות לפני שהיא מתחילה להרוג תהליכים.

האופציות "-r" ו"-h" מתפרשות כ"reboot" ו"halt" בהתאמה, ככה שהם מבצעות shutdown לאיתחול או לכיבוי המערכת.

root יכול גם להשתמש ישירות בפקודה shutdown. פקודה זו יכולה לשמש גם למחשב מקומי וגם למחשב מרוחק, אבל משתמשים בה בעיקר לכיבוי מרוחק כשהמקלדת המקומית אינה זמינה כך שאינך יכול להשתמש ב <alt><ctrl>. היא יכולה להועיל גם אם תוכנית תוקעת ומנטרלת את המקלדת.

לדוגמא:

```
telnet name_of_machine_with_no+operable_keyboard
[login as a user]
su
[give password]
```

קעת או שתבצע more | ps axu, מצא את ה id ששייך לתהליך של הפקודה הפוגעת בפלט של ps ובצע:

```
kill pid_of_offending_process
```

או תבצע איתחול מחדש למחשב שלך עם:

```
/sbin/shutdown -rn now
```

פקודה זו תבצע כיבוי מהר מאוד, ותדלג על פרוצדורת הכיבוי (הארוכה יותר) – שימושי כשהמערכת ממש מבוגבגת (האופציה -n תגרום ל"shutdown" להרוג את כל התהליכים לפני האיתחול המחודש).

שים לב בבקשה שמסיבות שקשורות לאבטחה, אינך יכול להיכנס למכונה אחרת כroot (דרך telnet). אתה חייב להיכנס למערכת כמשתמש ואז לבצע su לתת סיסמא ואז להפוך לsuperuser (root).

הפקודה shutdown יכולה גם לבצע כיבוי מאוחר יותר (כ root):

```
/sbin/shutdown -r 23:59
```

תאתחל את המחשב דקה לפני חצות. אני יכול להשתמש גם ב:

```
/sbin/shutdown -r +1
```

שאתחיל את המערכת דקה מעכשיו. אני יכול לבטל כיבוי מתוזמן בעזרת:

```
/sbin/shutdown -c
```

אם הפקודה shutdown ארוכה מדי עבורך, ייתכן ותרצה להשתמש ב שתי הפקודות האלו, שייבצעו בדיוק את מה שהשמות שלהן מציעים (כ root):

```
reboot  
halt
```

דרך קלאסית לכבות את המחשב שלך תהיה להעביר את המערכת שלך ל runlevel 0 (halt) או ל runlevel 6 (reboot) נסה את זה על ידי (כ root):

```
init 0
```

```
init 6
```

המשמעות של runlevels השונים מוסברת בקובץ /etc/inittab וכאן.

3.1.6 איך אני מתמודד עם תוכנית תקועה?

תוכניות מבוגגות אכן נתקעות תחת לינוקס. קריסה של אפליקציה לא אמורה, בכל מקרה, להשפיעה על מערכת ההפעלה עצמה כך שאינך צריך לאתחל את המחשב שלך לעיתים תכופות מדי.

שרתי לינוקס ידועים ככאלה שיכולים לרוץ במשך יותר משנה ללא איתחול מחדש של המערכת. מהנסיון שלנו, מערכת הפעלה שמתנהגת בצורה לא טובה יכולה להיות סימן לבעיות חומרה או קונפיגורציה:

באופן חוזר ונשנה נתקלנו בבעיות חימום ייתר של מעבד פנטיום (המעבד או המאוורר לא פעלו במהירות שהם היו אמורים לפעול בה או שהם עצרו, חריצי האיוורור שעל גבי הפנטיום היו ספוגים בליכלוך), ציפים גרועים של זיכרון, תיזמונים שונים של ציפים שונים של זכרון (אתה יכול לנסות לשנות את הסדר של הציפים, זה יכול לעזור), כיוון לא נכון של BIOS (אתה יכול לכבות את כל ההגדרות ה"מתקדמות", לינוקס מטפלת בדברים בעצמה). הודעת השגיאה "signal 11" קשורה בדרך כלל (99%) עם בעיות חומרה וסביר להניח שהיא תופיע כשאתה תבצע משימות מורכבות ואינטנסיביות על גבי המחשב: התקנת לינוקס, קומפילציית קרנל וכו'. אם לפנטיום שלך יש נטייה לחימום ייתר (נפוץ מאוד בפנטיומים מוקדמים), לפניך מספר טיפים כדי לשמור עליו קר, במיוחד בזמן מזג אויר חם: נקה את חריצי האיוורור שך המעבד, החלף את המאוורר של המעבד, תפעל את המחשב ללא המכסה וכוון מאור פנימה הגבר את מצב (wait state) בביוס, הקטן עומס מיותר, אל תבצע over clocking החלף את שומר המסך הסופר מפואר שלך במסך ריק.

לא באמת תקועות. חלק מהתוכניות יכולות ליצור רושם של תקועות, בעוד במציאות הן רק מחכות לקלט מהמשתמש. בדרך כלל, זה קורה כשהתוכנה מחכה לשם קובץ כארגומנט של שורת הפקודה ואף שם קובץ לא ניתן לה על ידי המשתמש, אז התוכנית נכנסת למצב ברירת המחדל שלה לקלט סטנדרטי (שהוא הקונסולה). לדוגמא, הפקודה הזו:

```
cat
```

יכולה להיראות כאילו היא תקועה בעוד למעשה היא מחכה לקלט מהמקלדת. נסה ללחוץ על `<ctrl>d` (שמשמעותו "סוף החיים") כדי לראות שזה יספק את הפקודה `cat`. דוגמא נוספת: ראיתי הרבה שאלות בקבוצות גיון לגבי הפקודה ה"מבוגבת" `tar` ש"נתקעת" בזמן שמנסים לפרוס קובץ למשל:

```
tar -zxv my_tar_file [worng!]
```

גם זה מחכה לקלט מהמשתמש, מאחר והאופציה `-f filename` לא צויינה, השם `"my_tar_file"` לא הוכר כשם קובץ. הפקודה הנכונה היא:

```
tar -zxvf my_tar_filename
```

שים לב בבקשה ששם הקובץ חייב להופיע מייד אחרי האופציה `"f"` (שמייצגת `"filename"`). זה לא יעבוד (טעות נפוצה מאוד):

```
tar -zxfv my_tar_file [worng]
```

כל תוכנית (תקועה או לא) יכולה להיהרג.

תוכנית במצב גרפי שרצה בחזית יכולה להיהרג על ידי לחיצה על `<ctrl>c`. זה לא יעבוד עם אפליקציות גדולות שחוסמות את `<ctrl>c`, כדי שבטעות לא יישתמשו בשילוב הזה כדי לפגוע בהם. עדיין אתה יכול לחזור לשליטה על ידי שליחת התוכנית לרקע על ידי שימוש ב `<ctrl>z` (אין ערובות שזה יעבוד) או החלפה לטרמינל אחר, על ידי שימוש ב `<ctrl><alt><f2>` למשל וכניסה למערכת בשם אותו משתמש שתקע את התוכנית (זה אמור לעבוד תמיד). ברגע שאתה שוב בעיניינים, מצא את בתוכנה שאתה רוצה להשמיד, לדוגמא:

```
ps
```

פקודה זו מתפשרת כ `"print status"` ומראה את רשימת התוכניות שמורצות כרגע על ידי המשתמש הנוכחי. בפלט של `ps`, אני מוצא את מזהה התהליך (`process id=PID`) של התוכנית שנתקע וכעת אני יכול להרוג אותה. לדוגמא:

```
kill 123
```

תהרוג את התוכנית עם מזהה התהליך (PID) `"123"`.

כמשתמש, אני יכול להרוג רק את התהליכים ששייכים לי (אלו שאני התחלתי). root יכול להרוג כל תהליך. כדי לצפות ברשימה של כל התהליכים שרצים במערכת הקלד:

```
ps axu | more
```

זה מציג את כל התהליכים שרצים כרגע (האופציה "a"), אפילו את אלו ללא הטרמינל (האופציה "x"), וביחד שם שם הלוג אין של המשתמש שלו שייך כל תהליך ("u"). מאחר וסביר להניח שהתצוגה תהיה ארוכה יותר מן המסך, השתמשתי באופציה "more" כדי שהתצוגה תעצור לאחר כל מסך מלא.

לפקודה kill יש את קיצור הדרך killall כדי להרוג תוכניות לפי שם, לדוגמא:

```
killall netscape
```

שתהרוג כל תוכנית עם "netscape" בשם שלה, בעוד

```
killall pppd
```

קרוב לוודאי תנתק כל חיבור חיוג על ידי הריגת השירות pppd (daemon).

לתוכניות מבוססות x-windows אין טרמינלי שליטה ויכולות להיות הכי קלות להריגה על ידי שימוש בפקודה זו (מוקלדת בטרמינל x):

```
xkill
```

כשמקלידים את הפקודה הזו, סמן העכבר הופך למשהו שדומה לגזר דין מוות; ואז כשאתה מצביע על החלון של התוכנית אותה רוצה להרוג ומקליק על הכפתור הימני בעכבר החלון נעלם והתוכנית המקשורת אליו מחוסלת.

קיצור דרך אל הפקודה האחרונה הוא לחיצה על <ctrl><alt><esc>, ושוב אותו דבר לחיצה על החלון בעזרת הסמן עם גזר הדין והתוכנית המקשורת נעלמת יחד עם החלון.

אם מערכת x-windows שלך קורסת באופן כזה שהיא אינה יכולה להשתקם, או שאתה פשוט נתקע, זו יכולה להיות הכי קל להרוג את שרת הא על ידי לחיצה על <ctrl><alt><backspace>. לאחר מכן, ייתכן וזה יהיה רעיון טוב להריץ את ps axu, כדי למצוא תוכניות x שעדיין רצות, ולהרוג אותן. אם לא תעשה את זה, ובאמת יש תוכניות שמתנהגות לא טוב שגרמה למערכת x-windows שלך לקרוס, היא עלולה לגרום לצרות שוב.

אם יש לך תוכניות ברקע, מערכת ההפעלה תמנע מבעדך לצאת מן המערכת עם הודעה כמו "there are stopped jobs". כדי להתגבר על זה ולצאת בכל זאת, כל מה שעליך לעשות זה לחזור על הפקודה logout או exit-- התוכניות שברקע יושמדו באופן אוטומטי ואתה תיצא החוצה.

קבצי ליבה.

core files

כשתוכנית קורסת, היא הדרך כלל "זורקת" core לתוך ספריית הבית שלך. תהליך זה מלווה בהודעה מתאימה. "core" הוא העתק של זיכרון (בתוספת debugging) שנועד להיות כלי ל debugging.

אם אתה משתמש שאינו מעוניין לעשות debugging לתוכנית, אתה יכול פשוט למחוק את coren :

```
rm core
```

או לא לעשות דבר (במידה ויזרק core חדש הוא ייכתוב על גבי הישן). אתה יכול גם לבטל את זריקת coren על ידי הפקודה:

```
ulimit -c 0
```

בדוק אם זה עובד על ידי:

```
ulimit -a
```

(זה מראה את ה"user limits", האופציה -a מתפרשת כ "all").

כדי להפוך את ביטול זריקת coren לקבוע לכל המשתמשים, ערוך את הקובץ /etc/profile (root) כשהולך ulimit מכוון. היכנס שוב למערכת כדי שהשינויים ל /etc/profile ייכנסו לתוקף.

אם אתה רוצה לראות כיצד ניתן להשתמש בקובץ core, (בספרייה שבה יש לך קובץ core):

```
gdb -c core
```

פקודה זו מפעילה את debugger של GNU על קובץ coren בשם "core" ומציגה את שם התוכנית שיצרה את coren הזה, את הסיגנל שבו תוכנה זו הושמדה, וכו'. הקלד quit כדי לצאת מהdebugger.

כדי ללמוד על המשמעות של הסיגנלים השונים, נסה את:

```
cat /usr/include/bits/sugnum.h | more
```

3.1.7 אפשרויות פקודה

רוב הפקודות מקבלות "אופציות" רבות. ניתן להציג אופציה עם "-" (מקף). לדוגמא:

`dir -l`

מראה את רשימת הספרייה הנוכחית בפורמט ארוך (פורמט ברירת המחדל הוא קצר). ניתן להציג כמה אופציות בשתי דרכים מקבילות:

`dir -l -a`

או

`dir -la`

כל אחת מהפקודות למעלה תציג לי רשימה של הספרייה הנוכחית בפורמט של קובץ ארוך (האופציה -l) ותכלול את כל הקבצים ברשימה, גם את הקבצים הנסתרים (האופציה -a).

רוב האופציות הפופולריות מסומנות באות אחת. זה ממשיך את באופן מסורתי את הדרך שבה משתמשים באופציות ביוניקס. ישנו גם סטייל חדש, שנראה כך:

`dir --help`

כאן, אורכה של אופציה אחת הוא יותר מאות אחת, והיא חייבת להיות מוצגת עם שני מקפים.

הפקודה למעלה מציגה עזרה זריזה לפקודה דיר, כולל רשימה של כל האופציות. מכיון שיש הרבה כאלו (יותר ממסך מלא) הייתי מקליד אותה כך:

`dir --help | more`

3.2 משתמשים, סיסמאות, הרשאות קבצים, ואבטחה

3.2.1 ספריות בית, root, הוספת משתמשים

המקום (הכמעט) יחיד על גבי הדיסק הקשיח אליו משתמשים (שאינם root) יכולים לכתוב הוא ספריית הבית שלהם, שהיא `/home/user_login_name`.

ספריית הבית מיועדת לכל הקבצים של המשתמש: הגדרות, קבצי קונפיגורציה של תוכניות, תיעוד, נתונים, המטמון של הדפדפן, דואר, וכו'. כמשתמש, אתה יכול ליצור תת ספריות תחת ספריית הבית שלך כדי לארגן את עצמך. משתמשים אחרים אינם יכולים לקרוא את הקבצים שלך או לכתוב אל ספריית הבית שלך, אלא אם כן תיתן להם רשות לעשות את זה.

משתמשים רגילים יכולים לראות, לכתוב ולבצע קבצים רבים אחרים במערכת (בנוסף לספריית הבית שלהם), אבל בדרך כלל הם אינם יכולים לשנות או להסיר (למחוק) אותם.

root (נקרא גם super user) הינו חשבון אדמיניסטרטיבי מיוחד, בעל היכולת לשנות כל קובץ במערכת. אין זה מומלץ לעבוד כroot על המערכת באופן שוטף--אם לא תנהג כך יש סיכוי שתשלם מחיר גבוהה על הטעויות שלך. הקם לך חשבון משתמש רגיל לעבודה יומיומית, חשבון נוסף לבנך, ואחד נוסף לאישתך.

החשבון root הוא בדרך כלל החשבון היחיד שקיים במערכת לאחר ההתקנה (אם לא יצרת חשבונות נוספים במהלך ההתקנה עצמה-א.א.).
לכן עליך ליצור חשבונות משתמשים לעבודה רגילה בלינוקס באופן יזום.

חשבון משתמש יכול להיווצר על ידי root כך למשל:

```
adduser joe
passwd joe
[הקלד את הסימא למשתמש joe]
```

[הקלד שוב פעם את הסימא למשתמש joe כדי להימנע מטעויות]

לכן קודם כל נכנסתי למערכת כroot. אחר כך, בשורת הפקודה, הקלדתי את הפקודה "adduser" עם הפרמטר "joe". זה יצר את החשבון "joe" בלינוקס שלי. לאחר מכן, הקלדתי את הפקודה "passwd joe" כדי לשנות את הסימא למשתמש "joe" למשהו קצת מאובטח.

כעת אני יכול לומר ל"joe" מהי הסימא שלה, ואז היא יכולה להיכנס למערכת ולשנות את הסימא שלה לפי מה שמתאים לה. שים לב שגם שם המשתמש וגם הסימא הינם case-sensitive.

root יכול לשנות את הסימאות של כל המשתמשים, למרות שהוא/היא אינם יכולים לקרוא אותן. [הסימאות מוצפנות, על ידי one way alogrithm ורק הגירסא המוצפנת מאוחסנת על גבי המערכת, בקובץ `/etc/passwd` (מערכות ישנות) או ב `/etc/shadow` (מערכות חדשות יותר, הגירסא ה"פתוחה" של הסימא לעולם אינה מאוחסנת. כשאתה נכנס למערכת, הסימא שאתה מקליד מוצפנת שוב באותו אלגוריתם (one way) ומושווה אל הגירסא המוצפנת שמאוחסנת ב `/etc/passwd` או `/etc/shadow`].

ההפרדה בין האדמיניסטרטור לבין המשתמש הופכת את מערכות לינוקס למאובטחות ועמידות--היא אפילו הופכת את החיים קשים יותר לזירוסים תחת לינוקס (התוכנית שמשתמש מריץ יכולה לכתוב רק אל תוך הספריות שלו ולכן אינה יכולה להשפיעה על חלקים חיוניים במערכת ההפעלה).

הנוהג הוא, שמשתמש משנה את הסיסמא שלו מייד לאחר הכניסה הראשונה שלו למערכת, לדוגמא:

```
passwd
(current) UNIX password: pass_OLD
new UNIX password: pass_NEW
retype new UNIX password: pass_NEW
```

במציאות הסיסמא לא תופיע על המסך בזמן שאתה מקליד אותה (מסיבות שקשורות לאבטחה). קח את הזמן אם אתה משנה את הסיסמא שלך בפעם הראשונה--זה יכול להיות קשה להקליד על "עיוור".

בלינוקס, אותה סיסמא משמשת ל:
כניסה לטרמינל טקסט.
כניסה לממשק גרפי.
שיחורר מנעילה של טרמינל טקסט.
שיחורר מנעילה של שומר מסך מוגן סיסמא בממשק גרפי.

3.2.2 קצת על אבטחת סיסמא

סיסמאות חלשות הינם בדרך כלל המקור השכיח ביותר לבעיות אבטחה. אפילו בבית, אתה עלול לחשוף את עצמך לצרות רציניות בגלל שמישהו יכול לפרוץ אל המחשב שלך בזמן שאתה גולש באינטרנט ולמחוק/לקרוא את הקבצים שלך, או להשתמש במחשב שלך כדי לעשות משהו באמת מרושע למחשב של תחנת המשטרה המקומית. לכן, שמור את כל השמות/סיסמאות שלך מאובטחים, אפילו בבית. ברגע שמישהו ייכנס אליך למחשב, (אפילו כמשתמש רגיל), הוא יכול להשיג גישה root די בקלות (תלוי עד כמה המערכת שלך מתוחזקת/מעודכנת +עד כמה הוא האקר טוב).

לפניך כמה דוגמאות לסיסמאות מסוכנות:
- שום סיסמא(אפשרי!)
- המילה "password" (וואו, זו באמת אחת חלשה)
- שם הכניסה שלך (שם הלוג אין והסיסמא זהים? המממ...)
- השם הפרטי שלך או השם הפרטי של בתך, בנך, בעלך, אשתך, של החברה שלך, או כל שם פרטי אחר. מספר השמות הפרטיים הוא די מוגבל.
- שם המשפחה שלך או כל שם משפחה אחר. מספר שמות המשפחה מוגבל באופן מפתיע!
אתה יכול לבדוק בנתונים של מפקד התושבים האחרון שנעשה בארה"ב ולהיות מופתע מהעובדה ששם המשפחה המשפחה ה"נדיר" שלך מהאי אבהמה מיוצג בצורה מכובדת מאוד בין 89,000 שמות המשפחה השכיחים ביותר בארה"ב.

<http://www.census.gov.genealogy.html>

- שם החיבה של הכלב שלך, של האישה שלך, של התוכי הקנרי שלך או של המחשב שלך (בני אדם משתמשים במבחר מצומצם מאוד של שמות חיבה, בוודאי שפחות משמות פרטיים).

- השם של קבוצת הספורט האהובה עלייך, סלבריטי, משחת השיניים שלך.
הזהר משמות של קבוצת כדורגל פופלרית כמו מאש. כנ"ל לגבי להקות רוק (מוזיקה).
- תאריך הלידה שלך, מספר תעודת זהות וכו'; רצף של ספרות יכול להיחשף בקלות.
- שם החברה שלך, מחלקה, קבוצת עבודה וכו'.
- סיסמא שכתובה על לוח השנה שלך או בצד המחשב.
- סיסמא שמשמשת אותך במקומות ציבוריים נוספים, למשל חנות באינטרנט או רשימת תפוצה. באופן כללי עלייך להשתמש בסיסמאות שונות למקומות שמנוהלים על ידי אירגונים שונים.

- כל מילה שנמצאת במילון האנגלי. המילון האנגלי לא מכיל כזו כמות של מילים כמו שזה נראה. האקר לא כל כך מיומן יכול להכין תוכנה שתצפין את כל המילים במילון (100,000? זה פחות מ1MB!) ולאחר מכן להשוות את כל המחרוזות המוצפנות אל מול הסיסמא המוצפנת שלך. נסה את התוכנית crack בעצמך כדי לגלות עד כמה זה קל.
מילות שבועה, ביטויי שפה "מגניבים" הופכים את הסיסמא לפגיעה מאוד לפיצוח.
- כל מילה אחרת, שם משפחה, שם פרטי, שם חייט מחמד, ולא משנה באיזו שפה לקראקר, כיסוי רוב השפות זו רק קצת יותר עבודה במידה והוא כבר כיסה שפה אחת. כמה שפות משמעותיות יש בעולם? 40? הקראקר רק תופס עוד כמה קבצים ומוסיף אותם אל רשימת הפיצוח שלו. הנקודה כאן היא שסך כל המילים שבני אדם בדרך כלל משתמשים בהן נמוך מגבול הקומבינציות הרנדומליות שניתן ליצור באופן תאורטי משילוב של תווים.
- כל ה"ל" בשילוב של אות/מספר בסוף או בתחילת הסיסמא כמו "yuoping1" למשל, יוצר באמת סיסמא חלשה.

סיסמא טובה צריכה להיות ארוכה באופן יחסי (מינימום 6 תווים, כמה מומחים ממליצים אפילו על מינימום של 10 תווים), שמכילה שילוב של אותיות (גדולות וקטנות, אם אפשרי), ספרות ותווים מיוחדים, ושמתנה באופן די קבוע (8-16 שבועות?).

למרבה הצער, ככל שהסיסמא טובה יותר, כך קשה יותר לזכור אותה. אני פותר את הבעייה הזו באופן כזה, אני לוקח לעצמי 10 דקות כדי להמציא את ה"סכמה" של הסיסמא שלי. נאמר, אני תמיד מתחיל ומסיים את הסיסמא שלי עם "שטרודל" (@), ומשתמש בשתי מילים מחוברות ל סימן קריאה, כשהאות האחרונה בכל מילה היא אות גדולה, משהו כמו: "@whutE!houSE@". נראה כמו סיסמא ראויה, שקל לזכור אותה ברגע שאני יודע מה החוק של הסיסמא שלי. אם אתה בעל זיכרון גאוני, אתה יכול לשקול סיסמאות מצויינות באמת שמיוצרות על ידי mkpasswd ()):

מנהל המערכת, האדמיניסטרטור, יכול לקבוע מדיניות סיסמא (אורך מינימלי, דרישה לתווים מיוחדים, תאריך תפוגה) דרך כלי שכלול בתוכנית הקונפיגורציה הבאה (root):

linuxconf

תחת התפריט "user account - policies - password & account policies" משתמשים רגילים לא יוכלו לקבוע לעצמם סיסמא קצרה מדי, מילה שמופיע במילון, או שאינה מכילה את כמספר המומלץ של תווים שאינם מן האלף בית (root יכול לשנות כל סיסמא לכל מה שעולה בדעתו, אבל יקבל אזהרה).

(לא מצאתי כל זכר לlinuxconf תחת RH 8.0. הפקודה redhat-config-users מפעילה את redhat user manager. מישוה המליץ לי להשתמש בwebmin, אכן שווה, לשיקולך-א.א)

בנוסף וודא שלכל קובץ שמכיל כל סיסמא שלך (/root/.kde/share/config/kppprc/) ישנן ההרשאות המתאימות ככה שאף אחד אחר לא יוכל לקרוא אותו. למשל, קורב לוודאי שתרצה:

```
chmod 600 kppprc
```

אם אתה משתמש בחיבור אינטרנט אנלוגי דרך טלפון רק למשך כמה שעות בשבוע, אתה יכול להסתדר אפילו עם סיסמא חלשה יחסית במערכת. אבל חשוב על אבטחת המערכת שלך אם אתה משתמש במודם כבלים, adsl, או אם אתה מחובר לאינטרנט למשך פרקי זמן ארוכים.

רוב משתמשי המחשב הסטנדרטיים משתמשים בסיסמאות חלשות להדהים. "קרוב ל-50 אחוזים ממשתמשי המחשב **מבוססים** את הסיסמאות שלהם על שם של קרוב משפחה, שותף לחיים או חיית מחמד. שלושים אחוזים משתמשים בשם של אליל הפופ או הספורט שלהם" כך מדווחת רשת CNN

[/http://www.cnn.com/2002/TECH/ptech/03/13/dangerous.passwords](http://www.cnn.com/2002/TECH/ptech/03/13/dangerous.passwords)

שים לב בבקשה לדגש על **מבוססים**. הוספת סיפורה למילה פשוטה כמעט שאינה הופכת את הסיסמא למאובטחת יותר.

3.2.3 שכחתי את הסיסמא של root

אפילו שאיני שוכח סיסמאות בכלל, הייתי לומד את הנושא הזה בפירוט בכל זאת מכיוון שהוא יכול לתת לי רמז לגבי איך אמא שלי עלולה לקרוא את היסטוריית שיחות הצ'אט שלי ICQ. :-)

שיטה ראשונה. הדרך הקלה ביותר לפתרון בעיית ה"שכחתי את הסיסמה" היא לאתחל את לינוקס בsingle-user mode, כל מה שעלייך לעשות הוא פשוט להקליד את הפקודה הבאה בזמן promptn של LILO :

```
linux single
```

זה יהפוך אותך לroot בלי לבקש ממך סיסמא. כעת כשאתה root, אתה יכול לשנות את הסיסמא של root על ידי שימוש בפקודה (אין צורך לדעת את הסיסמא הישנה) :

```
passwd
```

אם זה נשמע לך לא מאובטח, זה בגלל ששום מערכת הפעלה אינה מאובטחת אם לאנשים יש גישה פיזית אל החומרה שלך. אני באופן אישי לא אהבתי את פירצת האבטחה ש"linux single יצר במחשב הביתי שלי וסתמתי אותה על ידי הוספת השורות הבאות לקובץ /etc/lilo.conf (בסוף של "image=")

```
password="my_password"  
restricted
```

[הסיסמא של "LILLO" דרושה למצבים בהם מישהו מקליד את המילה "linux" עם פרמטר כל שהוא prompt של lilo בזמן האיתחול (איתחול רגיל ללא כל פרמטרים יהיה עדיין אפשרי ללא כל סיסמא)]. כדי שהשינויים שנעשו ב /etc/lilo.conf ייושמו, אני חייב להריץ מחדש את הפקודה lilo. מכיון שסיסמת הול של lilo אינה מאובטחת, עליי להפוך את /etc/lilo.conf לקריא אך ורק על ידי root:

```
chmod 600 /etc/lilo.conf
```

[הערת המתרגם: ניתן ליישם *linux single* גם אם אתה משתמש ב*grub* ולא ב*lilo*. העיניין מצריך קצת אקרובטיקה אבל בסך הכל עובד באופן די פשוט. בדקתי את זה וזה עבד, עשיתי את זה ככה:

1. במידה והתקנת את *grub* עם סיסמא, הקלד *p* והכנס את הסיסמא.

2. בחר את גרסת הלינוקס שלך והקלד *e* כדי לערוך.

3. בחר בשורה שמתחילה ב*kernel* ושוב, הקלד *e* כדי לערוך את השורה.

4. בסוף השורה הקלד רווח ואז *single*. לחץ על *ENTER* כדי לצאת ממצב העריכה.

5. חזור אל תפריט מערכות ההפעלה ואתחל את לינוקס. המערכת תאותחל אל *runlevel*

המבוקש. --א.א.]

שיטה שניה. דרך נוספת לפתרון בעיית "סיסמת root האבודה" היא לאתחל את המחשב שלך מדיסק (פלופי) האיתחול של לינוקס או מה *cd* של לינוקס. ואז, למצוא את המחיצה של root על גבי הדיסק הקשיח, להמיר אותה, ולערוך את הקובץ */etc/shadow*. (אני יכול לעשות את זה מכיון שלאחר איתחול מהפלופי אני נהיה root בלי שאתבקש להקליד סיסמא). בקובץ הסיסמא, אני מוחק את הסיסמא המוצפנת לroot (על ידי שימוש בעורך *pico* למשל), כך שלבסוף היא ריקה.

מידע על חשבון משתמש נמצא בקבצי הטקס הלא מוצפנים: */etc/passwd* וב-
/etc/shadow.

הקובץ */etc/passwd* מכיל מידע קריא מאוד על כל החשבונות במחשב שלי. בכל שורה יש 7 שדות תחומים על ידי נקודותיים (זה אומר 8 פסקאות שמופרדות על ידי נקודותיים): שם *login*, האות "x", המזהה המספרי של המשתמש, המזהה הראשי של קבוצת המשתמש, שדה להערה (לדוגמא, השם המלא של המשתמש), ספריית ה *\$HOME* של המשתמש, שם המעטפת (הכוונה היא לתוכנית שרצה ב*login*).

האיזון של המידע לגבי חשבונות במחשב שלי מאוחסן בקובץ */etc/shadow*. קובץ זה מאובטח יותר מכיון שבאופן נורמלי רק root יכול לקורא אותו. בקובץ זה כל שורה מתארת מידע מוצל ("shadow" information) על כל חשבון משתמש, ומכילה 9 שדות תחומים

בנקודותיים: שם login, סיסמא מוצפנת, מספר הימים שחלפו מהשינוי האחרון של הסיסמא מאז 1 לינואר 1970, מספר הימים שבהם עדיין מותר לשנות את הסיסמא, מספר הימים שאחריהם חייבים לשנות את הסיסמא, מספר הימים לפני תפוגת הסיסמא על מנת להזהיר את המשתמש, מספר הימים לאחר תפוגת הסיסמא שבהם החשבון מבוטל, ושדה שמור.

ישנן מערכות יוניקס ולינוקס (ישנות יותר) שלא מכילות את הקובץ `/etc/shadow` ומאחסנות את הסיסמא המוצפנת של המשתמש בשדה השני בכל שורה בקובץ `/etc/passwd` (השדה שבמערכות חדשות יותר מכיל רק את האות x).

לדוגמא, השדה שמופיע אחרי `root` בקובץ `/etc/shadow` שלי נראה בערך ככה:

```
root:$1$BuPbmLAz$ 1G7 . evlChyqaEI0TIZp0F . :11071:0:99999:7:-1:-1: 134540356
```

ולאחר שהסיסמא נמחקה, הוא נראה ככה:

```
root: :11071:0:99999:7:-1:-1:134540356
```

כעת, לחשבון של `root` אין סיסמא, לכן אני יכול לאתחל את המחשב שלי ובשורת הכניסה (login prompt) להקליד `root` ובמקום סיסמא פשוט ללחוץ `ENTER` (ריק, ללא כל סיסמא). לאחר כניסה מוצלחת למערכת, אני מייד קובע סיסמא ל `root` על ידי הפקודה:

```
passwd
```

נראה שלמרות מחיקת הסיסמא מ `/etc/shadow`, ההפצה דביאן לא תאפשר לך להיכנס למערכת ללא סיסמא (אבטחה משופרת?). במקרה כזה, מה שעלייך לעשות זה להחליף את הסיסמא בקובץ `/etc/shadow` בסיסמא מוצפנת של חשבון אחר, שאת הסיסמא שלו אתה כן מכיר, לאחר מכן, אתה יכול להיכנס למערכת מכיוון שאתה יודע את הסיסמא.

שליחה של סיסמא מוצפנת בדואר אלקטרוני יכולה להיות דרך להקמת חשבון למישהו מרוחק: "אני מקים חשבון ftp על השרת שלי בשבילך. שלח אלי במייל את הסיסמא המוצפנת שלך". לאחר שקיבלת במייל את הסיסמא המוצפנת, אתה מכניס אותה בשדה המתאים ב `/etc/shadow`. כעת, המשתמש יכול להיכנס למערכת, מכיון שהוא יודע את הסיסמא אבל פרט אליו איש לא יודע אותה.

כדי ליצור "גישת פלופי" אל המערכת שלי הייתי צריך להתאמץ קצת יותר, חשבתי להריץ מערכת ללא פלופי כלל, אבל לרוע המזל, התקליטורים של לינוקס ניתנים לאיתחול. כיוונתי את רצף האיתחול בביוס ככה שהמערכת תאתחל מהדיסק הקשיח ולא מהפלופי או מה `cd`, והוספתי סיסמא לביוס. עדיין, ניתן לשבור סיסמאות ביוס די בקלות, אפשר להסיר את הסוללה הקטנה שנמצאת על גבי לוח האם ולאפס את ההגדרות האלו, או שמישהו יכול פשוט להסיר את הדיסק הקשיח ולחבר אותו אל מחשב אחר...

אני שוקל להתקין מערכת קבצים מוצפנת שזמינה ללינוקס כעת, אבל בהתחשב בכל הטירחה שקשורה בהקמה כזו, אולי אתפשר על נעילת החדר שלי וזהו. אם הכל נשמע לך פראנואידי מידי, זה כנראה באמת כך--זה רק מחדד את העובדה שיש מעט מאוד אבטחת מידע גם תחת לינוקס, אם לפורץ יש גישה פיזית אל החומרה שלך.

3.2.4 שכחתי את סיסמת המשתמש שלי

אם משתמש רגיל, שאינו root, שוכח את הסיסמא שלו, זו אינה בעיה נכיון ש root יכול לשנות כל סיסמא. לדוגמא (roots):

```
passwd barbara
```

ייבקש ממך להקליד סיסמא חדשה ל "barbara" (ללא צורך בידע של הסיסמא הקודמת). אם משתמש רגיל (שאינו root) מבקש לשנות את הסיסמא שלו, הוא עדיין ייצטרך להקליד את הסיסמא הישנה לפני כן. (זו תכונת אבטחה כדי שאיש לא ישנה את הסיסמא שלך במידה והשאררת את הטרמינל שלך עזוב).

3.2.5 ביטול או הסרת חשבון משתמש

ניתן לבטל חשבון משתמש באופן זמני או להסיר אותו לחלוטין מהמערכת.

כדי לבטל חשבון באופן זמני (לנעול), אין כל צורך לשנות את הסיסמא שלו. כל מה שעליך לעשות זה להניח "*" כוכבית בתחילת השדה השני (לפני הסיסמא המוצפנת) בקובץ /etc/shadow. ה"חשבון זה אסור להיכנס למערכת". כשתרצה לשחזר את החשבון, מחק את ה"חשבון" כוכבית וחשבון המשתמש ייחזור לפעולה, עם הסיסמא הישנה שלו.

הנה דוגמא מ הקובץ /etc/shadow עם הסיסמא המבוטלת של "peter":

```
peter:*$1$narMEFm6$fhA1puOU422HiSL5aggLI/:11193:0:99999:7:-1:-1:134539288
```

יכולתי גם לנעול את חשבון המשתמש עם הפקודה הבאה:

```
passwd peter -l
```

ולשחרר אותה בעזרת:

```
passwd peter -u
```

כדי להסיר חשבון משתמש באופן בלתי הפיך מהמחשב, אני מבצע את המהלכים הבאים:
- נכנס למערכת כ root
- משנה את הזהות שלי למשתמש שאני מעוניין להסיר, כדי לבדוק אם ישנו דואר חשוב:

```
su doomed_user_login_name  
mail  
logout
```

- מוחק את חשבון המשתמש ואת הקבוצה שלו.

```
userdel doomed_user_login_name  
groupdel doomed_user_login_name
```

מסיר את הקשר של המשתמש לכל הקבוצות האחרות:

```
usermod -G doomed_user_login_name doomed_user_login_name
```

- כופה מחיקה של ספריית הבית של המשתמש עם כל התכולה שלה כולל כל תת הספריות:

```
rm -fr /hom/ doomed_user_login_name
```

3.2.6 יש לי בעייה עם הרשאות קבצים, הרשאת קבצים ובעלות על קבצים, איך זה עובד?

לינוקס (בדיוק כמו יוניקס) הינה מערכת הפעלה מרובת משתמשים מאובטחת, וזה יוצר רמה מסויימת של מורכבות עם "הרשאות קבצים". בעיות עם הרשאות קבצים יכולות להוביל לצרות נבזיות ולא צפויות. הבנת הרשאת קבצים חיונית לניהול כל מערכת הפעלה מרובת משתמשים (יוניקס, NT, או לינוקס) העצה שלי תהיה:

למד את מערכת הרשאות הקבצים של לינוקס (או של כל יוניקס) ; אתה לא תצטער על זה.

בעלות על קבצים. כל קובץ (או ספרייה) שייכים לבעלים (בדרך כלל שם login) ולקבוצה.

הבעלים הוא בדרך כלל האדם שיצר (או שהעתיק) את הקובץ. הקבוצה בדרך כלל מכילה אדם אחד--הבעלים, ויש לה שם זהה לשם של הבעלים, אבל זה לא חייב להיות ככה. קובץ יכול להימחק אך ורק על ידי הבעלים של אותו קובץ, או על ידי חבר בקבוצה שהקובץ נמצא בבעלותה, או על ידי root. משתמשים אחרים, בכל אופן, יכולים למחוק או לשנות את תוכן הקובץ אם ניתנה להם הרשות לכך. הבעלים והקבוצה שבבעלותה הקובץ ייתגלו מהפלט של הפקודה `ls -l` ("list in the long format"). לדוגמא, הפקודה הבאה:

```
ls -l junk
```

תפיק את הפלט הזה על המסך שלי:

```
-rwx----- 1 yogin inca 27 Apr 24 14:12 junk
```

זה מראה את הקובץ "junk", שנמצא בבעלות "yogin" והקבוצה "inca".

הבעלות על הקובץ ניתנת לשינוי על ידי שימוש בפקודות `chown` (change owner) ו-`chgrp` (change group), שבדרך כלל מבוצעות על ידי root:

```
chown peter junk
```

```
chgrp peter junk
```

```
ls -l junk
```

לאחר ביצוע 3 השורות שלמעלה, הפקודה `ls -l junk`, מפיקה את הפלט הזה על המסך שלי:

```
-rwx----- 1 peter peter          27 Apr 24   20:27 junk
```

שינוי בעלות על קובץ הופכת שימושית אם אתה מזיז/מעתיק קבצים כroots לשימוש על ידי משתמשים אחרים.

הרשאות קבצים. כעת, בעלים של קובץ יכול להפוך את הקובץ לנגיש בשלושה מצבים:

read (r), write (w) & execute (x)

לשלוש רמות של משתמשים :

owner (u), members of a group (g), others on the system (o)

אתה יכול לבדוק את הרשאות הגישה הנוכחיות על ידי שימוש ב:

```
ls -l filename
```

אם הקובץ נגיש לכל המשתמשים (owner, group, other) בכל שלושת המצבים (read, write, execute) הוא יראה:

```
-rwxrwxrwx
```

דלג על המקף "-" הראשון (הוא מראה "d" לספריות, "-" לקבצים נורמליים, "l" ללינקים, "c" לcharacter devices, "b" ל block devices, "p" ל named pipes, "f" ל stacks) לאחר תו זה, השלישייה הראשונה מראה את הרשאות הקבצים לבעלים על הקובץ, השלישייה השנייה מראה את הרשאות לקבוצה שבבעלותה הקובץ, השלישייה השלישית מראה את הרשאות למשתמשים אחרים. הרשאה שלילית נראת כ "-". לפניך פלט מפקודת `ls -l` לקובץ שבבעלות root, שבו לבעלים יש את כל הרשאות, אבל לקבוצה ולאחרים יש רק הרשאות קריאה וביצוע:

```
drwxr-xr-x 2 root root          21504 Apr 24 19:27 dev
```

האות הראשונה "d" מראה שהקובץ הוא למעשה ספרייה.

אתה יכול לשנות את הרשאות על קובץ שבבעלותך על ידי שימוש בפקודה `chmod`. לדוגמא, פקודה זו תוסיף את הרשאה לקרוא את הקובץ "junk" לכל. (=user+group+others)

```
chmod a+r junk
```

בפקודה למעלה, במקום "a" (=all), יכולתי להשתמש ב"u", "g" "o" (= user, group, others). במקום "+" (=הוספת הרשאה), יכולתי להשתמש ב"-" או ב"=" (הסר את ההרשאה או קבע את ההרשאה). במקום "r" (=read), יכולתי להשתמש ב"w" או ב"x" (=execute או write=).

דוגמא שנייה, פקודה זו תסיר את ההרשאה לבצע את הקובץ "junk" מאחרים:

```
chmod o-x junk
```

במקום אותיות, אתה יכול להשתמש גם במספרים כדי לציין את ההרשאות. על מנת להבין איך זה עובד, הסתכל בזה:

```
execute=1  
write=2  
read=4
```

סך ההרשאות למשתשים הוא סכום השלושה:

0 = שום הרשאה בכלל (לא לכתוב, לא לקרוא, או לבצע) (שכיח)

1 = לבצע בלבד (לא נפוץ)

2 = לכתוב בלבד (לא נפוץ)

3 = לכתוב ולבצע (לא נפוץ)

4 = לקרוא בלבד (שכיח)

5 = לקרוא ולבצע (שכיח)

6 = לקרוא ולכתוב (שכיח)

7 = לקרוא, לכתוב ולבצע (שכיח)

ההרשאה לכל שלושת הרמות של המשתמשים (בעלים, קבוצה, אחרים), מושגת על ידי הדבקה של שלוש הספרות יחד, אחת אחת. לדוגמא, הפקודה:

```
chmod 770 junk
```

תיתן לבעלים ולקבוצה את מירב ההרשאות, אבל ללא הרשאות לאחרים. הפקודה:

```
chmod 666 junk
```

תיתן לכל שלושת רמות המשתמשים (בעלים, קבוצה, אחרים) את ההרשאות לקרוא ולכתוב

(אבל לא לבצע) את ומהקובץ "junk". שים לב בבקשה ל"666". זהו שילוב די נפוץ, לפחות לגבי אדם אחד שאני מכיר, זוהי ההוכחה שליניקס (וכל יוניקס לצורך העניין) היא עבודה של השטן 0:-> .

פקודה זו:

```
chmod 411 junk
```

תיתן לבעלים את ההרשאה לקרוא בלבד, ולקבוצה ולשאר המשתמשים לבצע בלבד את הקובץ. זה לא נראה מועיל כל כך, אבל יכול להיות מצחיק, לפחות למשתמשי ליניקס מצפון אמריקה שמחייגים 411 (מספר טלפון) לעזרת מרכזנית.

הצורה המספרית להצגת הרשאות קבצים נקראת "octal" בגלל שלמספרים יש הבסיס 8 (הבסיס של המערכת הדצימלית הוא 10). הספרה הגבוהה ביותר המערכת האוקטלית הוא 7 (למערכת האוקטלית יש 8 ספרות : 0-7, כלמערכת הדצימלית יש 10 ספרות : 0-9). התצוגה באוקטלית היא סימון נח לייצוג הבינארי של הרשאות קבצים, כשכל הרשאה מסומנת ל"set" או ל"denied" עם אחד או אפס והסך הכל מיוצג כרצף של אפסים או אחדים, כמופיע בדיאגרמה הבאה:

users				
class:	owner	group		
others				
exemple permissions:	rwX	rw -	r - -	
absent permissions:	- - -	- - X	- wx	
binary representation of the permissions:	111	110	100	
octal representation of the binary:	7	6	4	

הרשאות לספריות.

הרשאות לספריות הינן בעלות משמעות שונה מאשר הרשאות לקבצים "רגילים". לקבצים רגילים:

z=הרשאה לקרוא את תכולת הקובץ

w=הרשאה לשנות את תוכן הקובץ

x=הרשאה לבצע את הקובץ.

לספריות:

z=הרשאה להציג את רשימת הקבצים בסיפרייה

w=הרשאה ליצור או למחוק קבצים בתוך הספרייה

x=הרשאה לגשת אל הספרייה.

מכל בחינה אחרת הרשאות נקבעות לספריות ממש כמו לקבצים רגילים.

ברירת המחדל של הרשאות קבצים עם umask.

כשקובץ חדש נוצר, ניתנות לו הרשאות ברירת מחדל. במערכת שלי הן:

- rw - r- -r - -

זה אומר שקבצים שנוצרו על ידי משתמש יכולים להיקרא ולהיכתב על ידי המשתמש הזה; הקבוצה והאחרים יכולים רק לקרוא את הקובץ. עדיין, במערכת הרדהאט שלי, כברירת מחדל, משתמשים אינם יכולים לקרוא את התוכן של ספריות הבית של משתמשים אחרים מכיון שההרשאות של ספריות הבית הן:

```
drwx - - - - -
```

אני יכול לבדוק את הרשאות ברירת המחדל לקובץ שזה עתה נוצר על ידי שימוש ב:

```
umask -S
```

(האופציה "-S" מתפרשת כ "symbolic" ואומרת ל umask להציג את ההרשאות באופן נוח לקריאה, המקום מצב התצוגה הדיפולטי המספרי)

אני יכול לשנות את הרשאות ברירת המחדל לקבצים החדשים שלי על ידי שימוש בפקודה כמו:

```
umask u=rw,g=,o=
```

שתיתן לבעלים את הרשאות הכתיבה והקריאה לקבצים חדשים שיווצרו (r+w), ושום הרשאה לקבוצה ולאחרים.

שימוש במספרים כדי לקבוע הרשאות ברירת מחדל עם umask הוא עניין קצת יותר מורכב. המספר מראה את ההרשאות שאתה שולל ממשתמשים (בניגוד ל chmod). לכן:

```
umask 000
```

ייתן הרשאות מלאות לכולם על קבצים חדשים שיווצרו. הדוגמא הבאה תיתן הרשאות כתיבה וקריאה לבעלים, ואפס הרשאות לכל השאר (וזוה כנראה מה שתרצה לעשות):

```
umask 177
```

כדי להפוך את השינויים לקבועים לכל המשתמשים במערכת, התאם את השורה (שורות) הרלוונטית בקובץ /etc/profile.

3.2.1 נגן mp3 שלי נחנק. הצליל לא נשמע טוב (איך להגדיר suid)

יכול להיות שלנגן mp3 אין די כח מעבד (הוא דורש הרבה). יכול להיות שהמחשב שלי פשוט גרוע. או שיכול להיות שאתה מריץ יותר מדי תוכניות שדורשות כח עיבוד רב, בבת אחת. או, מה שסביר להניח שקורה, אתה צריך להריץ את הנגן עם עדיפות גבוהה יותר. (העדיפות של תוכנית יכולה להיקבע בעזת הפקודה nice--בדוק את man nice או את info man). נסה להפעיל את הנגן roots--לתוכניות שמופעלות על די root ניתנת עדיפות גבוהה יותר מאשר לאלו שמופעלות על ידי משתמשים רגילים. אם זה פותר את הבעיה, קבע את ה "suid" לכל התוכניות ככה שלכל המשתמשים תינתן "זוהות משתמש אפקיבית" לגבי הבעלים של הקובץ (בדרך כלל root) כמשפעילים אותן, לדוגמא:

```
chmod a+s /usr/bin/xmms
```

ייעשה את הטריק לתוכנית xmms. הפלט של:

```
ls -l /usr/bin/xmms
```

במחשב שלי הוא כעת:

```
-rwsr-sr-x 1 root root 908k Feb 22 2000 /usr/bin/xmms
```

ה "s" הראשונה מצביעה על כך שה `suid=substitute-user-id` קבוע-מוגדר. ה "s" השנייה מצביעה על כך שה `sgid=substitute-group-id` גם כן קבוע-מוגדר. לכן לכל מי שמבצע את xmms ניתנת הזכות האפקטיבית של הבעלים של התוכנה וזכות הקבוצה של הקבוצה שבבעלותה התוכנית, אשר בדוגמה למעלה שניהם root.

הגדרת suid לתוכנית יכולה להוות פירצת אבטחה במערכת שלך. זהו אינו המקרה ברשת ביתית סגורה וכשמגדירים suid לתוכנית שהמקור שלה ניתן למעקב. בכל מקרה, אפילו בבית, לא הייתי מגדיר suid לחתיכת קוד שהמקור שלו אינו וודאי, אפילו שהוראות ההתקנה האיצו בי לנהוג כך. בנוסף, זה בהחלט רעיון רע מאוד להגדיר suid לתוכניות רבות המערכת שלך--זה פוגע בכל הרעיון שעומד מאחורי האבטחה של יוניקס.

בכל מקרה, ישנן כמה תוכניות שדושרות suid לתיפקוד תקין, kppp (חיבור המודם הפופולרי תחת kde) למשל. זה נובע מהצורך בגישה ישירה לחומרה--לפעמים רק לroot מותר לעשות את זה.

אם אתה נתקל בבעיות חוזרות לגבי תיפקוד "חלק" של המערכת שלך, או שחלק מהחומרה שלך (צורב למשל) נוטה לקרוס, נסה להפחית את מספר ה daemons במערכת הלינוקס שלך. הרץ roots את הפקודה setup (פקודה של RH) ובטל את כל ה"שירותים" (services) שאתה לא באמת צריך. לחילופין, אתה יכול לעבור לשורת פקודה, לכבות את הממשק הגרפי (הפקודה 3 roots init), מה שיגרום לביצועי המערכת שלך להשתפר ללא ספק.

לאילו שצריכים (אוהבים) מערכת הפעלה "אוניברסלית" (תחנת עבודה, שרת, מחשב משרדי, קופסת משחקים, מולטימדיה, וכו', והכל בעת ובעונה אחת), ישנם טלאים מיוחדים לקרנל של לינוקס: "low latency patch" & "pre-emptive kernel patch" שתוקפים באופן אגרסיבי את בעיית ה"latency" (זמן איחזור) שמערכות עמוסות מדי מציגות.

הבא: תיזמון משימות בעזרת "&", "at", "batch" ו-"cron"

3.3 תזמון משימות עם "&", "at", "batch", ו-"cron"

3.3.1 כיצד אני מבצע פקודה "ברקע"?

על ידי שימוש ב"&" בסוף הפקודה. למשל, זה יתחיל את licq (קליינט של icq) בטרמינל של x ברקע, ככה שלאחר החלת הפקודה, ה x טרמינל שלי אינו חסום:

```
licq &
```

מספר הזיהוי של התהליך, job_number, מודפס על המסך, כך שאתה יכול להשתמש בו עם פקודות שקשורות אליו. הפקודות הן:
fg job_number (= "foreground", הבא את התהליך שרץ ברקע חזרה אל התצוגה/שליטה המיידית שלי, התחל אותו מחדש אם הוא נעצר).
bg job number (= "background", שלח את התהליך לרקע, התחל אותו מחדש אם הוא נעצר, בדיוק כאילו הוא הותחל על ידי שימוש ב"&").
z <ctrl> (שלח את התהליך שרץ כרגע בחזית לרקע ועצור אותו).
Jobs (הצג רשימה של עבודות פעילות).
Kill process_ID (השמד את התהליך, השתמש בפקודה ps כדי למצוא את המזהה של התהליך אותו מעוניין להשמיד).

כדי לגרום לתהליך שרץ ברקע להמשיך לרוץ לאחר שאתה מתנתק, אתה יכול להשתמש בפקודה nohup (= no hangup) למשל:

```
nohups make &
```

3.3.2 איך אני מבצע פקודה בזמן מסויים (על ידי שימוש ב "at" או "batch")?

הפקודה "at" תבצע פקודה (דות) שאתה מציין בזמן ותאריך לפי בחירתך. למשל, אני יכול להתחיל לנגן מוזיקה מכונן התקליטורים שלי בשעה 7 בבוקר:

```
at 7:00
```

```
cdplay<ctrl>d
```

בדוגמא למעלה, הכנסתי את השורה "at 7:00" בשורת הפקודה ואז לחצתי על ENTER. לזה, הפקודה at הגיבה בהצגת פרומפט, ">at". שם הקלדתי את הפקודה "cdplay" ואז לחצתי על ctrl d, לחצתי ENTER, ה">at" הבא הופיע, שבו הייתי יכול להכניס את הפקודה הבאה שתבוצע מייד לאחר "cdplay", גם כן ב 7:00.
וכן הלאה, יכולתי לתזמן כך הרבה פקודות אחת אחרי השנייה. לאחר שהקלדתי את הפקודה האחרונה, סיימתי את הקלט שלי עם ctrl d. חשוב על ctrl d כמה ששולח "סוף החיים" לקלט הנוכחי. אל תלחץ על ctrl d פעמיים מכיון שזה יוציא אותך החוצה—זה מה ctrl d עושה כשמשתמשים בו ישירות בשורת הפקודה של לינוקס.

אתה יכול להציג את העבודות שתיזמנת לביצוע על ידי שימוש ב:

at -l

שייתן לך רשימה ממוספרת של עבודות שמחכות.

במידה ושינת את דעתך אתה יכול להסיר עבודה מהרשימה, לדוגמא:

atrm 8

תסיר את העבודה עם המספר 8 ברשימה.

יכולתי גם לתזמן עבודה לביצוע בשלב מאוחר יותר, לדוגמא:

at 23:55 12/31/00

startx

תתחיל את מערכת ה x-windows שלי בדיוק בזמן למילניום החדש (5 דקות לפני חצות ב31 בדצמבר 2000).

אם אינך יכול לבצע את הפקודה at, בדוק אם השירות (daemon) "atd" טעון (כroots, השתמש ב ntsysv). אם אינך יכול לבצע את הפקודה at כמשתמש רגיל למרות שהיא עובדת עם root, בדוק אם הקובץ הריק /etc/at.deny קיים ושאינ כל קובץ בשם /etc/at.allow. זו אמורה להיות הגדרת ברירת המחדל והיא מאשרת לכל המשתמשים לבצע את הפקודה at. אם אתה מעוניין שמתמשים מסויימים בלבד יוכלו להשתמש בat, צור קובץ בשם /etc/at.allow ומנה את המשתמשים הללו שם.

לאופציות אחרות, בדוק את:

man at

אם אתה מבקש לבצע עבודה שדורשת כח עיבוד רב ברקע בזמן שהעומס על המערכת נמוך, אתה יכול לבחור בפקודה batch. אני יכול, לדוגמא, להריץ את setiathome (תוכנית שמעבדת נתונים בכדי לעזור בחיפוש אחר אינטליגנציה חוצונית, SETI) על ידי שימוש ב:

batch

at>setiathome<ctrl>d

בדוגמא זו, הכנסתי את הפקודה batch ולאחר מכן את הפרומפט של "at>", הכנסתי את הפקודה שרציתי שתבוצע ברקע. העבודה מנסה להתחיל מייד, אבל מתקדמת רק כשהעומס נמצא מתחת ל 0.8 אתה יכול לבדוק את העומס של המערכת על ידי בחינת התוכן של הקובץ (הוירטואלי) ./proc/laodavg. למשל:

```
cat /proc/loadavg
```

כשקובץ batch מסיים, הפלט נשלח אלי דרך דואר אלקטרוני.

3.3.3 איך אני מגדיר את cron?

cron (תהליך של לינוקס שמבצע עבודה ברקע, בדרך כלל בלילה) מוגדר כברירת מחדל במערכת redhat שלך. כך שאינך צריך לעשות דבר לגביו אלא אם אתה מעוניין להוסיף משימות שיבוצעו במערכת שלך על בסיס קבוע או לשנות את הזמן בו cron מבצע את החובות המוטלות עליו.

שים לב בבקשה שייטכן וחלק מהעבודה ש-cron מבצע יכולה להיות חיונית לתיפקוד התקין של המערכת שלך לאורך זמן. בנוסף לדברים אחרים cron יכול:

- לבנות מחדש את קבצי מסד הנתונים שנמצאים בשימוש בזמן שאתה מחפש קבצים על ידי שימוש בפקודה locate.
- לנקות את הספרייה /tmp.
- לבנות מחדש את דפי ההסברים של לינוקס (manual pages).
- לעשות רוטציה לקבצי הרישום (log files), להיפטר מקבצי הרישום הישנים ביותר, לשנות את שמם של קבצי הרישום, וליצור קבצי רישום חדשים.
- לבצע בדיקות נוספות אחרות, כמו הוספת פונטים שהעתקת לאחרונה אל המערכת.

לכן, כיבוי מכונת הלינוקס שלך בלילה לא יהיה הרעיון המוצלח ביותר-- במקרה כזה לcron לא תהיה ההזדמנות לעשות את עבודתו. במידה ואתה מעוניין לכבות את המחשב שלך ללילה, יכול להיות שתרצה לכוון את cron ככה שהוא יבצע את החובות המוטלות עליו בזמן אחר.

כדי לבדוק מתי cron מתעורר כדי לבצע את עבודתו, הסתכל בקובץ /etc/crontab, לדוגמא:

```
cat /etc/crontab
```

ייטכן והוא יכיל משהו כזה:

```
# run-parts
01 * * * * root run-parts /etc/cron.hourly
02 4 * * * root run-parts /etc/cron.daily
22 4 * * * root run-parts /etc/cron.weekly
42 4 1 * * root run-parts /etc/cron.mothly
```

אתה יכול לראות שישנן ארבע קטגוריות של עבודות cron: שמבוצעות מדי שעה, מדי יום, שבוע, וחודש. אתה יכול לשנות אותן או להוסיף קטגוריה משלך. ככה זה עובד:

העמודות מצביעות על: דקות (0-59), שעות (0-23), היום בחודש (1-31), החודש בשנה (1-12), היום בשבוע (0-6 ראשון עד שבת). ה" " מתפרש כ"כל ערך אפשרי".

לכן, בדוגמא המצוטטת, העבודות שמכוונות לביצוע כל שעה, מבוצעות בכל פעם שהשעון של המחשב מראה "ודקה אחת", מה שמתרחש בכל שעה, דקה אחת לאחר השעה הנתונה. העבודות היומיות מבוצעות בכל פעם שהשעון מראה 2 דקות לאחר השעה 4, מה שקורה

פעם ביום. העבודות השבועיות מבוצעות 22 דקות לאחר השעה 4 בבוקר בימי ראשון. העבודות החודשיות מבוצעות 42 דקות לאחר השעה 4 ביום הראשון של כל חודש. הספרייה עם קובץ הסקריפט שמכיל את הפקודה (פקודות) שאמורות להתבצע, נראה ברשומה האחרונה בכל שורה.

אם אתה רוצה שהעבודות שלך יבוצעו בצהריים במקום ב4 בבוקר, שנה את ה4 ל12. cron מתעורר בכל דקה ובודק אם הקובץ /etc/crontab שונה ככה שאין צורך להפעיל מחדש משהו לאחר שביצעת את השינויים שלך.

אם אתה רוצה להוסיף עבודה אל הcron שלך, הנח סקריפט שמריץ את העבודה המבוקשת (או לינק אל הסקריפט שלך) בספרייה /etc/cron.hourly או /etc/cron.weekly או /etc/cron.monthly.

לפניך דוגמא לרשומה ב /etc/crontab שגורמת לעבודה להתבצע שלוש פעמים בשבוע (שני, רביעי, שישי):

```
02 4 * * 1, 3, 5 root run-parts /etc/cron.weekly
```

דוגמא שנראתה ב usnetd מראה איך לשלוח באופן אוטומטי במייל קובץ רישום:

```
Re: help in crontab
From: Dean Thompson <Dean.Thompson@csse.monash.edu.au> Date:2001-03-03 16:35
newsgroups: comp.os.linux.admin.networking,comp.os.linux.security
>How can I set the job mail abc@abc.com < /var/log
>every day in the /etc/crontab -e file?
You could try the following entry and see if you meet with any success:
0 0 * * * (/bin/mail abc@abc.com < /var/log/messages) > /dev/null 2>&1
```

shell 3.4

3.4.1 מה זה מעטפת (shell) והאם אני רוצה להשתמש באחרת?

מעטפת היא התוכנית שמפענחת את מה שאתה מקליד אל תוך שורת הפקודה ומחליטה מה לעשות עם זה. ניתן לעורר מעטפת גם בצורה שאינה אינטרקטיבית, למשל לבצע רשימה של פקודות שהוקלדו מראש והוכנסו אל תוך קובץ טקסט ("shell script" a). אתה יכול לחשוב על מעטפת כמקבילה של "command.com" מדוס (מפענח שורת הפקודה) ועל shell script כמקביל של קבצי batch בודוס (*.bat). בהשוואה לדודנים שלהם מדוס, המעטפת והסקריפטים נמצאים על סטרואידיים.

ישנן כמה מעטפות זמינות בלינוקס (במידה והתקנת אותן):

bash : ("bourne again" shell) sh, bourne shell, הסטנדרט במערכות יוניקס רבות),
csh (C shell, בעל תחביר דומה ל"c" שפת התיכנות, זמין ברוב מערכות היוניקס).
pdksh (public domain korn shell), tcsh (tiny C shell, נמצא בשימוש במערכות קטנות
בדרך כלל) sash (stand alone shell, ניתן לשימוש כשספירות אינן זמינות).
zsh, ash, וכנראה עוד כמה.

מעטפת ברירת המחדל במערכת שלי (וסביר להניח שגם בשלך) היא bash, שהינה מעטפת מצויינת וסטנדרטית, ואני לא מוצא סיבה משכנעת למה ניוביי כמוני ירצה לשנות את זה. bash תואמת לאחור באופן מלא עם bourne shell (המעטפת הכי פופלרית ביוניקס) ומכילה הרבה שיפורים ותכונות טובות ממעטפות אחרות. מנקודת מבט של משתמש מתחיל, המעטפות השונות כלולות בלינוקס מסיבות היסטוריות ובגלל תאימות לאחור של סקריפטים שיידרשו מעטפת מסויימת בכדי לרוץ. [ישנן מעטפות שיועילו אם אתה כותב תוכניות שמכוונות להתקנים מיוחדים ו"מוטמעים" (embedded), שייתכן ויריצו מעטפת קטנה (a" tiny" shell)].

אתה יכול לבדוק איזו מעטפת אתה מריץ על ידי שימוש ב:

```
echo $SHELL
```

אם אתה מעוניין לנסות מעטפת אחרת, הקלד, לדוגמא:

```
tcsh
```

מה שיגרום למעטפת c קטנה (tiny c shell) להתחיל לפעול. כשסיימת הקלד:

```
exit
```

שיחזיר אותך למעטפת הקודמת (שימוש בexit במעטפת הראשונה שלך יוציא אותך אל מחוץ למערכת-log out). אתה יכול לבדוק כמה מעטפות ערמת אחת על השניה על ידי

הצגת המשתנה הסביבתי של רמת המעטפת (shell level):

```
echo $SHLVL
```

בפקודה למעלה, ה\$ מתפרש כ"הרחב את הערך של משתנה המערכת הסביבתי",
"SHLVL" הוא שם המשתנה, ו"echo" היא פקודה שמדפיסה דברים.

המעטפת לכל משתמש מצויינת בשדה האחרון בקובץ הסיסמאות /etc/passwd.
אם אתה באמת רוצה לשנות אותה, ערוך (root) את הקובץ הזה והחלף את ה"/bin/bash"
במעטפת שבחרת.

3.4.2 כיצד אני מבצע התאמה אישית למעטפת שלי?

במכונה שלי שורת הפקודה נראית כך:

```
[stan@marie stan] $ _
```

כאן stan הוא השם איתו אני נכנס למערכת (log in), marie הוא השם של המחשב,
stann השני הוא השם של ספריית העבודה הנוכחית שלי, וה_ מייצג את הסמן.

שורת הפקודה (prompt) מוגדרת על ידי משתנה סביבתי שנקרא PS1. כדי להציג את
ההגדרות הנוכחיות שלי, אני יכול להשתמש ב:

```
echo $PS1
```

ההגדרות הכלל מערכתיות של הפרומפט (לכל המשתמשים במערכת) נמצאות בקובץ
/etc/bashrc שבמערכת שלי מכיל שורה כזו:

```
PS1="[u@h W] \$ "
```

כדי לבצע התאמה אישית לפרומפט, אני יכול לערוך את הקובץ /etc/bashrc (כ root)
ולהכניס כמעט כל טקסט בתוך סימני הציטוט ("). הנה המשמעות של כמה קודים מיוחדים
שאני יכול לבחור לשימוש:

- \u שם המשתמש של המשתמש הנוכחי (\$LOGNAME=),
- \h שם המחשב שמריץ את המעטפת (שם המארח),
- \H שמו המלא של המארח,
- \W הבסיס של השם של ספריית העבודה הנוכחית,
- \w השם המלא של ספריית העבודה הנוכחית,
- \\$ מציג "\$" למשתמשים רגילים ו"#" לroot,
- ! מספר ההיסטוריה של הפקודה הנוכחית,
- # מספר הפקודה הנוכחית (כפי שמבוצעת במעטפת הנוכחית),
- \d התאריך הנוכחי,
- \t השעה הנוכחית (24 שעות),
- \T השעה הנוכחית (12 שעות) - bahe 2.0 בלבד,
- \@ השעה הנוכחית (am-pm) - bahe 2.0 בלבד,
- \s שם המעטפת,

\a - אזעקת צליל (ביפ),
 \j - מספר העבודות שיש למשתמש,
 \n - שורה חדשה,
 \\ - בקסלאש,
 \[- התחל רצף של תוים לא ניתנים להדפסה,
 \] - סיים רצף של תוים לא ניתנים להדפסה,
 \hחח - תו ה ASCII שמתייחס אל המספר האוקטלי חחח,
 \$(date) - פלט מהפקודה date (או כל פקודה אחרת לצורך העניין),

לפניך דוגמא שמראה איך להוסיף צבע. הסתכל בפרק הבא לגבי פרטים על צבע:

```
PS=1" \[033[1;32m\] [\u@\h \W] \$\ [033[0m\] "
```

ישנה גם רמה שניה של פרומפט, היא נקבעת על ידי המשתנה שנקרא PS2. המעטפת משתמשת ברמה שניה של פרומפט כאשר היא מחכה לקלט נוסף. במערכת שלי הפרומפט המשני הוא ">". אני לא דואג יותר מדי לגבי PS2, אבל אם יכולתי הייתי מגדיר אותו כמו PS1. ישנם גם PS3 ו-PS4 אבל הם נראים לעיתים נדירות.

3.4.3 צבע בטרימינל טקסט

ניתן להפיק צבע בטרימינל טקסט על ידי שימוש ב "ANSI escape sequences". לדוגמא:

```
echo -e "\[033[44;37;5m ME \[033[0m COOL"
```

הנ"ל מגדיר את הרקע ככחול, את החזית כלבן, מסך מהבהב, ומדפיס ME, לאחר מכן מאפס את הטרימינל חזרה להגדרות ברירת המחדל שלו ומדפיס COOL. ה -e הוא אופציה ספציפית לפקודה echo--היא מאפשרת את הפיענוח של התוים המיוחדים. ה "[033" מייצג את escape sequences. ה m קובעת אטריביוט ולכן מסיימת את הרצף. הקודים הממשיים בדוגמא למעלה הם "44;37;5" ו-"0".

שנה את ה "44;37;5" כדי ליצור שילובי צבע אחרים--המספר/סדר של הקודים אינו משפיע. הקודים הניתנים לבחירה רשומים למטה:

אפס את כל האטריביוטים להגדרות ברירת המחדל שלהם	0
הגדר (bold) (a-bit oriented line discipline)	1
הגדר חצי-בהיר (בתצוגת צבע על צבע)	2
הגדר קו תחתון (בתצוגת צבע על צבע)	4
הגדר היבהוב	5
הגדר וידאו הפוך	7
הגדר בהירות נורמלית	22
כבה קו תחתון (underline)	24
כבה היבהוב	25
כבה וידאו הפוך	27
הגדר חזית שחורה	30
הגדר חזית אדומה	31

הגדר חזית ירוקה	32
הגדר חזית חומה	33
הגדר חזית כחולה	34
הגדר חזית לאדום ארגמן	35
הגדר חזית לחן (בין כחול לירוק)	36
הגדר חזית לבנה	37
הפעל קו תחתון (underscore)	38
כבה קו תחתון (underscore)	39
הגדר רקע שחור	40
הגדר רקע אדום	41
הגדר רקע ירוק	42
הגדר רקע חום	43
הגדר רקע כחול	44
הגדר רקע לאדום ארגמן	45
הגדר רקע לחן	46
הגדר רקע לבן	47
הגדר רקע ברירת מחדל	49

קודים מענינים אחרים:

נקה מסך	\033 [2j
נקה את כל נוריות הבקרה שעל המקלדת (לא יעבוד מטרמינל X)	\033 [0q
קבע את ה"scroll lock led"	\033 [1q
קבע את ה"num lock led"	\033 [2q
קבע את ה"caps lock led"	\033 [3q
הזז את הסמן לשורה 15, עמודה 40	\033 [15;40H
פעמון (ביפי)	\007

LED's (=light emitting diodes) הן הנוריות שעל המקלדת שמסמנות את <capslock>, <numlock> ו- <scrolllock> מופעלים.

הסתכל ב man console_codes לעוד.

3.4.4 כיצד אני מדפיס סימנים בקונסול, באפליקציות מצב טקסט, וב-X?

הפרוצדורה שמתוארת כאן יכולה לתת לי גישה מהירה אל סט התווים המורחב של ה־cpc שלי (קודים 128-255) והיא די נייחת בעולם ה־cpc: היא עובדת במייקרוסופט ווינדוס, דוס (אם יש לך מנהל התקן ANSI מותקן), ובתוך כל אפליקציות מצב טקסט בלינוקס (כולל מעטפת שורת הפקודה). מצאתי שזה שווה את הזמן אותו הקדשתי לשינון הקודים לתווים המעטים בהם אני נוטה להשתמש הכי הרבה.

זה עובד כך. וודא ש- <numlock> פועל. לאחר מכן לחץ על <alt> והחזק אותו לחוץ. בזמן ש- <alt> לחוץ, הקלד את 4 הספרות האלו בפד הספרות המרובע שבצד ימין על

המקלדת: 0181. כעת שחרר את <alt> והאות היונית mu "µ" תופיע. מצאתי כמועילים את התווים הבאים: ° 176 (מעלות), ± 177 (פלוס מינוס), ² 178 (בריבוע), ³ 179 (בשלישית), µ 181 (mu יוני), · 0183 (נקודת הכפל), è 232 (האקסנט הצרפתי agrave), é 233 (האקסנט הצרפתי aigu), ä 228 (ה a-umalaut הגרמני), ó 243 (ה u-zamknieten הפולני), ø 248 (ה o-barn הסקנדינבי) ü 252 (ה u-umalaut הגרמני).

כעת, אם אני באמת רוצה, אני יכול להיות הבעלים של קובץ בשם $\mu\text{m}\cdot\text{c}\pm\text{b}^3$. מייקרוסופט ווינדוס, ANSI של דוס, ויוניקוד (unicode) שונים במקצת מהתווים למעלה, אבל ה"ליבה" השימושית נשארת זהה. הסתכל ב: <http://www.hclrss.demon.co.uk/demos/ansi.html> אם אתה מעוניין לדעת את פרטי ההבדלים. לינוקס משתמשת בסטנדרט של יוניקוד.

תחת X, שילובי המפתח שלמעלה לא יעבדו. אבל אני יכול להשתמש ב:

kcharselect&

או ב:

gcharmap&

כדי לבחור תו יוניקוד ולהעתיק אותו אל האפליקציה שלי. לא כל תו יוניקוד זמינים עדיין, אבל רבים כן זמינים. מעמודי יוניקוד אחרים מאשר עמוד 0, ייתכן והתווים יוצגו או שלא, תלוי באפליקציה שלך ובזמינות של glyph בפונט שלך. לדוגמה, אני יכול בוודאות להשתמש בתווים הבאים ברוב האפליקציות שב KDE (אם הם מוצגים בדפדפן שלך, תלוי בדפדפן שלך ובזמינות של פונט יוניקוד מתאים):

יונית (עמוד יוניקוד 3, תווים 913-969)

רוסית (עמוד יוניקוד 4, תווים 1040-1103)

ורבים אחרים. אתה יכול למצוא תו יוניקוד נפוצים (מספריים) ואת ייחוס הhtml הסימבולי שלהם ("character entity") ב:

http://www.hclrss.demon.co.uk/demos/ent4_frame.html

3.4.5 איך אני כותב סקריפט מעטפת פשוט?

צור קובץ טקסט (ASCII) שיכיל את סקריפט המעטפת (shell script). לדוגמה, אני אשתמש בעורך pico כדי לכתוב סקריפט שיריץ את התוכנית tar עם כל הפרמטרים שבדרך כלל דרושים כדי לפרוס קבצי tarball שאני מוריד מהאינטרנט (אני אף פעם לא מצליח לזכור את האופציות של tar). החלטתי לקרוא לסקריפט שלי "untar":

pico untar

מכיון שהקובץ "untar" לא קיים בספרייה הנוכחית שלי, הוא נוצר על ידי עורך הטקסט pico.
כעת, אני מקליד את התוכן של הסקריפט שלי:

```
#!/bin/bash
echo this is the script file $0
echo untarring the file $1
# this calls tar with options -xvzf (extarct,
# verbose, filter through gzip, input filename)
tar -xvzf $1
```

אני שומר את הקובץ עם `<ctrl>o` ויוצא עם `<ctrl>x`

השורה הראשונה של הסקריפט, מתחילה ב"#!" (נקרא pound-bang), הינה מיוחדת--היא אומרת למעטפת באיזו תוכנית להשתמש כדי לפענח את הסקריפט שלי. בדוגמא זו, הסקריפט ייפוענח על ידי המעטפת `bash`: `/bin/bash`. השורה הראשונה חייבת להתחיל ב"#!" או שהסקריפט לעולם לא ירוץ (הקובץ ייפוענח כקובץ טקסט רגיל). שורות אחרות שמתחילות ב"#" הינן הערות למחבר (קוראים, משתמשים) המעטפת והן זוכות להתעלמות מוחלטת מצד המחשב.

ה `$0`, `$1`, `$2` ... בסקריפט שלי הם הפרמטרים שהועברו אל הסקריפט שלי. לדוגמא, אם אני מריץ סקריפט בשם "myscript" עם שבע פרמטרים כמו זה:

```
myscript a b c d e f g
```

אז `$0` יהיה בתוך "myscript" כבעל הערך "myscript", ל `$1` יהיה הערך "a", `$2` יהיה "b", `$3` יהיה "c", וכן הלאה.

בשורה השניה ושלישית בסקריפט הדוגמא שלי, הפקודה `echo` מדפיסה על המסך כל מה שעוקב אחרי אותה שורה, מרחיבה את `$0` ו-`$1` לערכי הפרמטרים שהועברו לסקריפט. השורות הרביעית והחמישית מכילות הערה שכתבתי לעצמי כדי להזכיר לעצמי מה ניסיתי להשיג, רק למקרה שאי פעם אצטרך לשנות את הסקריפט שלי. השורה האחרונה מבצעת את העבודה האמיתית.

ברגע שהסקריפט נכתב, אני הופך את הקובץ לבר הרצה לבעלים של הקובץ (`"u"=user`):

```
chmod u+x untar
```

והסקריפט שלי מוכן להרצה באופן הבא:

```
./untar my_tar.tar.gz
```

הסקריפטים של לינוקס בהחלט עשירים, גמישים, עוצמתיים ויכולים להיות מורכבים. בכל מקרה, אין צורך בידע מיוחד כדי לכתוב סקריפטים פשוטים לאוטומציה של משימות פשוטות. אתה רק צריך לאסוף ביחד קבוצה של פקודות שנמצאות בשימוש שכיח, אחת אחרי השניה, אל תוך קובץ. אני משתמש בסקריפטים בגלל שאני עצל מדי כדי להקליד את אותן קבוצות של פקודות שוב ושוב.

רצף פקודות פשוט באמת יכול להיות מוקלד אל תוך קובץ טקסט ולהיות מועבר אל תוך מעטפת לביצוע מידי על ידי שימוש ב:

```
source my_file
```

[אין צורך ב"pound bang" הראשוניים או בהרשאות הרצה.]

3.4.6 המשמעות של ציטוטים

בדרך כלל סימנים אלו מיוחדים במשמעותם לגבי המעטפת:

```
\ ' " ` < > [ ] ? | ; # $ ^ & * ( ) = <Space> <Tab> <Newline>
```

ישנם ארבעה סוגים שונים של ציטוטים: קו נטוי אחורה (\), מרכאות ('), מרכאות כפולות ("), מרכאות הפוכות (').

הקו הנטוי אחורה \ מתפרש כ: בטל את המשמעות המיוחדת של התו שבא אחרי.

ציטוט עם ' ' (שתי מרכאות בודדות, לא כפולות) מתפרש כ: צטט בדיוק, בטל כל תו מיוחד בתוך המרכאות.

ציטוט עם " " (שתי מרכאות כפולות) מתפרש כ: בטל את התוים המיוחדים בתוך המרכאות פרט ל \$ ' \

זוג ה " " (שתי מרכאות הפוכות) מתפרש כ: בצע תחליף לפקודה בתוך המרכאות תחילה. כך שמה שבתוך המרכאות הפוכות מבוצע על ידי המעטפת תחילה, ולאחר מכן הפלט מועבר אל הפקודה מחוץ למרכאות. אותו הדבר ניתן ליישום בעזרת \$ (command).

דוגמאות. אני יכול ליצור ספרייה מצחיקה שנקרת "*" על ידי \ ציטוט או " ציטוט:

```
mkdir \ *  
mkdir ' * '
```

זה מחביא את המשמעות המיוחדת של ה "*" מהמעטפת (ללא הציטוט זה ייתפרש כ"כל הקבצים בתוך הספרייה הנוכחית").

3.4.7 הפניה מחודשת של פלט/קלט

ישנם שלושה זרמים חשובים של פלט/קלט: קלט סטנדרטי ("standard input-"stdin), פלט סטנדרטי ("standard output-"stdout), ופלט שגיאה סטנדרטי ("standard error output-"stderr). הם כולם ברירת מחדל לקונסול ("קונסול"=המקלדת לקלט והמסך לפלט), אבל ניתן להפנות אותם מחדש.

בכדי להפנות מחדש פלט סטנדרטי אני משתמש ב ">". לדוגמא:

```
dir my_dir > filelisting.txt
```

יפנה את הפלט הסטנדרטי של הפקודה dir אל תוך קובץ הטקסט filelisting.txt ודבר לא אמור להופיע על גבי המסך שלי. ניתן לערוך את הקובץ (בעזרת pico filelisting.txt) או להטמיע אותו בתוך מסמך.

בכדי להפנות את השגיאה הסטנדרטית, אני צריך להשתמש במבנה ">2". לדוגמא:

```
dir my_dir 2> errorlisting.txt
```

הפקודה שלמעלה תשלח פלט נורמלי אל המסך שלי ושום דבר אל הקובץ אלא אם כן dir מייצר שגיאה. כשתתרחש שגיאה, דבר לא יופיע על המסך שלי, והקובץ errorlisting.txt יכיל את הודעת הגיאה, שיכולה להיות משהו כמו:

```
dir my_dir > file_and_error_listing.txt 2>&1
```

מה שיפנה תחילה את הפלט הסטנדרטי אל קובץ טקסט, ולאחר מכן תפנה את השגיאה הסטנדרטית אל מיקום זהה למיקום הפלט הסטנדרטי. קצת מעוות, אך שזה עובד, אבל זה עובד.

בדוגמאות למעלה, אם הקובץ (שאליו מתבצעת ההפניה) כבר קיים, האינפורמציה תיכתב עליו. כדי להוסיף לקובץ קיים, אני משתמש ב ">>" כמו בדוגמאות אלו:

```
dir my_dir >> filelisting.txt
dir my_dir 2>> errorlisting.txt
dir my_dir >> file_and_error_listing.txt 2>&1
```

אם תהיה לגבי הסימן ">2", כאן, בזריזות, רציונליזציה של העניין. לזרמים הסטנדרטיים יש מתארים (descriptors) סטנדרטיים. "0" הוא קלט סטנדרטי, "1" קלט סטנדרטי ו"2" שגיאה סטנדרטית.

```
dir my_dir > file.txt
```

זה קיצור ל

```
dir my_dir 1> file.txt
```

ולכן הדוגמא למטה מפנה את השגיאה הסטנדרטית:

```
dir my_dir 2> file.txt
```

ניתן גם להשתמש בסימן "ן" כדי לשלוח ("pipe") את הפלט מפקודה אחת כקלט לפקודה אחרת. בדוגמא פופלרית זו, הפלט מ dir נשלח ("piped") אל more (more=פקודה שעוצרת את התצוגה לאחר כל מסך מלא):

```
dir | more
```

ניתן גם לפצל את הפלט ככה שהוא יילך גם אל הקובץ וגם אל המסך על ידי שימוש ב"tee":

```
dir | tee filelisting.txt
```

הפקודה נקראת "tee" כאנלוגיה לאות-צורה-T שבה אינסטלטורים משתמשים, שמחלקת את הזרימה.

חלק זה עסק עד כה בהפניית פלט סטנדרטי. הפניית קלט סטנדרטי אינה מועילה כמו הפניית הפלט, אבל ניתנת לביצוע על ידי מבנה כזה:

```
cat < my_file
```

ישנו גם משהו שנקרא הפנייה בתוך שורה (in-line redirection) של הפלט הסטנדרטי, שמיושם בעזרת "<<". שכח מזה, זה נראה לי כחסר כל שימוש. בכל זאת, הנה דוגמא למקרה שאי פעם באמת תזדקק לזה (כאן, ה">" מתפרש כפרומפט משני):

```
cat << my_marker
> my_line_from_the_keyboard
> another_line_from_the_keyboard
> my_marker
[הסמן שבו אני בוחר מסיים את הפניית התוך שורה]
```

פרט להפנייה אל קבצים רגילים ו"פילטרים" (כמו שמראה הדוגמא למעלה), ניתן להפנות מאל התקנים וקבצים מיוחדים אחרים. כמה דוגמאות.

דוגמא להפנייה אל קובץ התקן. הדוגמא הבאה מציגה רשימה של קבצים בטרמינל הטקסט הרביעי:

```
dir > /dev/tty4
```

דוגמא להפנייה לקובץ "FIFO" מיוחד. פקודה זו שולחת את ההודעה "you are lucky" אל משתמש ה ICQ בר המזל 77777777 UIN (בהנחה שאתה מחובר אל שרת icq עם תוכנת licq שלך):

```
echo message 77777777 "you are lucky" > ~/.licq/licq_fifo
```

הנ"ל עובד מכיון שהקובץ "licq_fifo" בספריית licq שלך הוא קובץ תור (queue) "fifo" (first-in-first-out) מיוחד. כיצד הפקודה למעלה יכולה להיות שימושית יותר משליחת הודעה על ידי שימוש בממשק הגרפי היפה של licq? לדוגמא, אתה יכול לכתוב סקריפט קצר כדי להרשים משתמשי icq עמיתים בעזרת הודעות מרובות (זהות):

```
#!/bin/bash
echo Messaging UIN: $1 Message $2 Times: $3
# The next command puts your licq in the status
"on-line, invisible".
echo 'status *online' > ~/.licq/licq_fifo
c=0
while [ $c -le $3]
do
echo message $1 $2 > ~/.licq/licq_fifo
c='expr $c + 1'
echo $c " "
```



```
done
echo 'status offline' > ~/.licq/licq_fifo
echo "all done"
```

הדוגמא שלמעלה נותנת לך רעיון לגבי איך ניתן להשתמש ב־licq לאוטומציה, בזכות מודל התקשורת החכם של licq (עם קובץ fifon) והפניית קובץ פשוטה.

3.4.8 תוי מעטפת מיוחדים (metacharacters)

בדרך כלל לתוים אלו יש משמעות ציחדת כלפי המעטפת:

= * & ^ \$ # ? [] () <newline> <tab> <space> ; | < > " ' \

הנה המשמעות של כמה מהם:

\ " ' משמשים לציטוט.

< > משמשים להפנייה מחדש של קלט/פלט.

| מעביר (pipes) את הפלט של הפקודה שמשאל לסימן pipen " | " אל הפקודה שמימין לסימן pipen.

; מפריד פקודות מרובות שנכתבות בשורה אחת.

<space> -ו <tab> מפרידות את מילות הפקודה.

<newline> משלים את הפקודה או סט של פקודות.

() סוגר פקודה או פקודות כדי שישוגרו במעטפת נפרדת (subshell) למשל (dir) .

{ } סוגר קבוצה של פקודות כדי שישוגרו על ידי המעטפת הנוכחית. [dir] . הרווח דרוש.

& גורם לפקודה הקודמת להתבצע ברקע (באופן אסינכרוני, כשהליך נפרד ועצמאי) ככה שהפקודה הבאה לא תחכה לסיימה.

* כששם קובץ

?

[]

&& הינם "AND" שמחברים שתי פקודות.

command1 && command2 ייבצע את command2 רק אם command1 תצא עם סטטוס היצאה 0 (ללא שגיאה). לדוגמא: cat file1 && cat file2 יציג את file2 רק אם התצוגה של file1 הצליחה.

|| הוא "OR" שמחבר שתי פקודות.

command2 || command1 ייבצע את command2 רק אם commadn1 יוצאת עם סטטוס היציאה ללא אפס (עם שגיאה). לדוגמא: cat file2 | cat file1 יציג את file2 רק אם file1 לא הצליח.

= מעניק ערך למשתנה.

לדוגמא: פקודה זו:
me=blahblah

מעניקה את הערך "blahblah" למשתנה שנקרא "me". אני יכול להדפיס את שם המשתנה על ידי שימוש ב:

```
echo $me
```

\$ מקדים את שם המשתנה שיורחב.

המשתנים מוענקים על ידי שימוש ב "=" או הינם אחד מהמשתנים המוגדרים מראש (שלא ניתן להעניק אליהם):
\$0 שם המעטפת או המעטפת שמבוצעת.
\$# מספר פרמטרי המיקום אל הפקודה.
\$1 הערך של הפרמטרים המוצבים המועברים אל הפקודה. \$2 הוא הפרמטר המיוחס השני המועבר אל הפקודה. וכו' עד \$9.
\$* מרחיב את כל הפרמטרים המוצבים שמועברים אל הפקודה.
@\$ מרחיב את כל הפרמטרים המוצבים שמועברים אל הפקודה, אבל מצוטטים ביחידות כש"\$@" נמצא בשימוש.
הסתכל ב man bash אם אתה באמת זקוק ליותר.

3.5 התקנת חבילה ומנהל החבילות rpm?

3.5.1 איך אני מתקין תוכנית שהורדתי מאינטרנט?

התשובה תלויה בסוג החבילה שהורדת. אתה יכול להימנע מהרבה כאבי ראש שקשורים בהתקנה אם תוריד תוכניות בצורת חבילות הבינארי של רד האט rpm*. (זה הפורמט שאני בוחר אם ניתנת לי האפשרות לבחור).

התקנת חבילות הבינארי (binary) של רדהאט

* אם התוכנית שאני מעוניין להתקין היא חבילת בינארי של רדהאט (rpm.*). אני יכול או להשתמש בשורת הפקודה, או בכלי ממשק גרפי. אני אוהב להשתמש בכלי של שורת הפקודה בגלל שהוא מהיר וחופשי מבעיות. כלי ההתקנה מנהל החבילות של רדהאט נקרא rpm. ראשית אני קורא מידע על תוכן החבילה (אופציונלי):

```
rpm -qpi ny_new_file.rpm
```

זה מתשאל (המצב "q", חייב להיות האות הראשונה אחרי המקף) את החבילה שעדיין לא הותקנה (האופציה "p") כך שהיא מציגה את המידע (האופציה "i") שהחבילה מכילה. אם אני מעוניין להתקין את הבילה, אני מריץ (roots):

```
rpm -ihv my_new_file.rpm
```

הפקודה למעלה מבצעת את עבודת ההתקנה. היא מריצה את rpm ואומרת לו להתקין את החבילה (המצב "i", חייב להיות האות הראשונה לאחר המקף) בזמן שהיא מדפיסה על גבי המסך מידע נוסף מהרגיל (האופציה "h" = מציגה "hashes" כדי להראות את התקדמות הפריקה, האופציה "v" = verbose). התכנים של החבילה מופצים אל הספריות שאליהן הם שייכים (rpm יודע להיכן הם שייכים). לאחר התקנה זו, התוכנית מוכנה להרצה, אני רק צריך לדעת את שם התוכנית (executable) ואת מיקומה. אם אני לא מצליח למצוא את התוכנית, זה מציג רשימה של הקבצים שהחבילה מכילה ביחד עם ספריות היעד שלהם:

```
rpm -qp1 my_new_file.rpm
```

פקודה זו מתשאלת (האופציה "q") את החבילה שעדיין לא מותקנת (האופציה "p") ככה שהיא מציגה (האופציה "i") את כל הקבצים שהחבילה מכילה.

כלי הממשק הגרפי (GUI) של rpm הם: gnoptro (הגירסה הישנה, שהגיע עם רדהאט 6 מבלבלת אבל הגרסאות החדשות משופרות הרבה יותר), kpackage (זמין רק עם ההפצות האחרונות), glinti (הישנה איטית מאוד, הגיע עם רדהאט 5.2).

טיפול בבעיות. rpm אמור להיות מנהל תוכנה אינטליגנטי. אם ההתקנה נכשלת אני קורא את הודעת השגיאה ואולי יכול לחשוב מה לעשות:
(1) ההתקנה נכשלה בגלל שיש ברשותי גירסה קודמת של אותה חבילה והגרסאות מתנגשות. פתרון: אל תתקין, אלא שדרג את החבילה.

(2) ההתקנה נכשלה בגלל שיש צורך בחבילה אחרת לפני כן. עליי למצוא את החבילה החסרה ולהתקין אותה קודם, ואז לנסות ולהתקין שוב את החבילה המבוקשת. במקרים קיצוניים, אני יכול לבחור להתעלם מהחבילות שבהן החבילה שלי תלויה (dependencies) (אני באמת צריך לדעת מה אני עושה כאן כי ייתכן והתוכנה תחדל מלתפקד כראוי):

```
rpm -ivh --nodeps my_new_file.rpm
```

או אולי אפילו:

```
rpm -ivh --nodeps --force my_new_file.rpm
```

התקנה מsource-code tarball

* במידה ומה שהורדתי מהאינטרנט הוא קוד מקור של לינוקס בצורת טרבל דחוס (*.tar.gz, *.tgz), פרוצדורת ההתקנה ארוכה יותר וקצת יותר מורכבת מאשר rpm בינארי בלבד. אני בדרך כלל מתקין את התוכנית roots.

ראשית, אני משנה את ספריית העבודה הנוכחית שלי ל/usr/local:

```
cd /usr/local
```

שנית, אני פורס את קובץ הטרבל שהורדתי מהאינטרנט:

```
tar -xvzf /home/the_dir_where_the_tarball_is/my_tarball.tar.gz
```

זה פותח (האופציה "x") את התוכן של הטרבל *.tar.gz (או *.tgz), מצבע עליו unzip (האופציה "z"), בזמן שהוא מדבר אלי יותר מן הרגיל (האופציה "v" verbose). בבקשה שים לב שהאופציה "f" מתפרשת כקובץ (file), ככה ששם הקובץ עוקב מיד אחרי האות "f". התכנים של הטרבל נפרסים אל תוך תת תיקיה שנקרא tar יוצר תחת ספריית העבודה הנוכחית שלי, שבמקרה הזה היא /usr/local/. הטרבל יודע כיצד צריכה להיקרא הספרייה החדשה.

אם הטרבל אינו דחוס (*.tar), אני יכול להשתמש ב:

```
tar -xvf /home/the_dir_where_the_tarball_is/my_tarball.tar
```

שלישית, עליי לפענח כיצד נקראת הספרייה החדשה, ואז אני עושה cd אליה:

```
dir
cd _the_new_program_subdir
```

מאחר לכמה מבספריות יש שמות ארוכים, אני משתמש באופציה ההשלמה האוטומטית כדי לחסוך בהקלדה--אני מקליד רק את הכמה אותיות הראשונות ואז לוחץ על <tab>.

רביעית, רוב התוכניות מקומפלות על ידי ביצוע שלושת הפקודות האלו:

```
./configure
make
make install
```

ייתכן וייקח לפקודות למעלה כמה זמן (דקה? חצי שעה?) להשלים את עבודתן. אם אחת מהן נכשלת, יהיה זה רעיון טוב לקרוא את הREADME או את INSTALL או מה שלא מגיע עם התוכנית החדשה. ייתכן וחלק מהתוכניות יידרשו קוסטמיזציה של הסביבה (הגדרת הנתביב שלהן) או התקנת ספרייה נוספת, או משהו אחר. זה יכול לעצבן לפעמים. ייתכן ותוכניות פשוטות מאוד לא ייצטרכו את /configure. / או את "makeinstall", יכול להיות שרק "make" יספיק.

חמישית, אם הכל הולך כשורה, אני מוצא את התוכניות החדשות שקומפלו. שמן של התוכניות מופיע בירוק כשאני מריץ את הפקודה הזו:

```
ls --color
```

כעת, אני יכול להריץ את התוכנית, לדוגמא:

```
./the_executable
```

כמה מהתוכניות יתקינו באופן אוטומטי את התוכנית אל /usr/local/bin, ככה שאני יכול לרצות ולנסות את:

```
/usr/local/bin/the_executable
```

שישית, אם אני מתכוון להריץ את התוכנית לעיתים קרובות, אני יוצר לינק סימבולי אליה מהספרייה /usr/local/bin:

```
cd /usr/local/bin
```

```
ln -s /usr/local/the_new_program_subdir/the_executable
```

באופן זה, התוכנית (למעשה, לינק סימבולי אליה) נמצאת בנתיב שלי וניתנת להרצה פשוט על ידי הקלדת שמה (אין צורך להקליד את הנתיב המלא אליה יותר). כמה תוכניות יתקינו את התוכנית (או לינק אליה) אל תוך הספרייה "bin" במקרה זה דלג על הצעד האחרון.

התקנה מחבילת rpm קוד מקור

ישנן גם תוכניות שמופצות כחבילות "source code rpm". הן דורשות התקנה של חבילת rpm*. ביחד עם כלי rpm כמתואר בחלק הראשון של פרק זה. אבל מאחר וה "rpm" מתקין את קוד המקור (בדרך כלל בקוד מקור של שפת "C"), עליי לקמפל את קוד המקור על ידי ביצוע הרצף: "make install ; make ; configure ; nake". כפי שעשיתי לקוד מקור שמופץ כטרבול (ראה תשובה קודמת).

עבור לפרק 4.1: נושאי איתחול (LILLO & GRUB)

