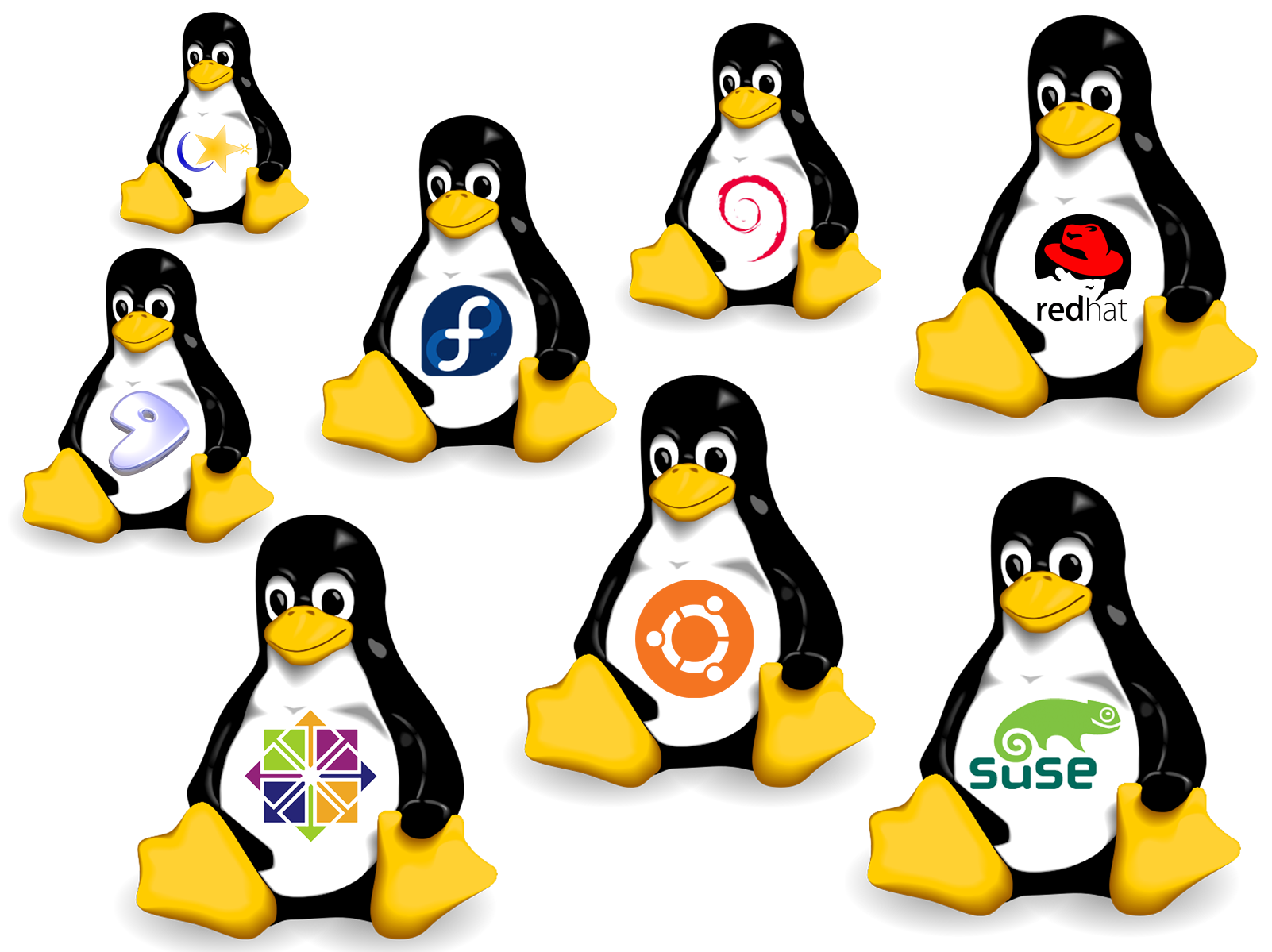
**מבוא ללינוקס**



תוכן

[לינוקס 3](#_Toc459107145)

[מה זה לינוקס? 3](#_Toc459107146)

[GNU ורישיון GPL 3](#_Toc459107147)

[הרצת לינוקס 4](#_Toc459107148)

[תקשורת שרת לקוח ב- Unix ו- Linux 8](#_Toc459107149)

[כניסה למערכת וזיהוי משתמש 9](#_Toc459107150)

[SHELL 10](#_Toc459107151)

[הכנסת פקודות באמצעות ה- Shell 11](#_Toc459107152)

[תווים מיוחדים 12](#_Toc459107153)

[פקודות לקבלת עזרה 13](#_Toc459107154)

[פקודות פשוטות 13](#_Toc459107155)

[Unix ו- Linux מערכת הקבצים 15](#_Toc459107156)

[שמות קבצים ו- Wildcards 16](#_Toc459107157)

[פקודות פשוטות על מערכת הקבצים 17](#_Toc459107158)

[מערכת ההרשאות וארגון הקבצים 23](#_Toc459107159)

[הרשאות (קיצורי דרך למתקדמים) 26](#_Toc459107160)

# לינוקס

## מה זה לינוקס?

לינוקס היא מערכת הפעלה “חופשית” בעלת קוד פתוח, שנוצרה במקורה על ידי לינוס טורבלדס, בשנת 1991.

לינוס, שהיה אז סטודנט פיני ממוצע, שיחק עם מכונה המריצה סוג של יוניקס המכונה minix.

לינוס החליט לשכתב את קוד המקור, ולכתוב קרנל (ליבה) משלו של מערכת שתהיה דמויית יוניקס.

על ההיסטוריה של יוניקס תוכל ללמוד בקישור :

<http://halemo.net/edoar/0017/0002.html> או <https://he.wikipedia.org/wiki/Unix>

הוא יצר את הליבה, ושיחרר אותה לעולם דרך האינטרנט, כשהוא פירסם את הליבה, הוא לא חשב על כך שהמערכת תהפוך למה שהיא הפכה, הוא ראה את זה כהובי ותו לא.

כיום המערכת היא מערכת שלמה ומלאה, שיכולה לעשות כל מה שמערכת הפעלה אחרת מסוגלת לעשות, ובצורה יעילה, זולה ומעל לכל, וכמובן כשהמערכת ממלאת את העיקרון של “תוכנה חופשית”

רוב מערכות ההפעלה היום מופצות במצב "סגור": אפשר להשתמש בהן אך לא ניתן לשפרן או לשנותן. לינוקס, לעומת זאת, מופצת עם קוד מקור פתוח: נגיש לכולם עם רישיון לשנות אותו. מהבחינה הזאת, מדובר במערכת הפעלה חופשית לחלוטין.  כל זאת תחת סייג חשוב אחד: אסור לשנות את לינוקס ולמכור אותה כמוצר סגור בלי לחשוף את קוד המקור.

ניתן להשיג תוכנות קוד פתוח, בהן דפדפן האינטרנט Firefox, הפועלות על גבי מערכות הפעלה בקוד סגור כמו Windows. בהקשר של לינוקס - המערכת מבוססת על קוד פתוח ורוב התוכנות המיועדות עבורה מבוססות על קוד פתוח והן מופצות חינם או במחיר סמלי.

### GNU ורישיון GPL

**GPL – General Public License**

תוכנה חופשית

מתוך (<http://www.gnu.org/philosophy/free-sw.html>)

*"Free software" is a matter of liberty, not price. To understand the concept, you should think of "free" as in "free speech", not as in "free beer*".

*Free software is a matter of the users' freedom to run, copy, distribute, study, change and improve the software*

* *הזכות להריץ את התכנית לכל מטרה*
* *הזכות ללמוד איך התכנית עובדת ולהתאים אותה לצרכיך*
* *הזכות להפיץ את התוכנה*
* *הזכות לשפר את התוכנה ולהפיצה עם השינויים לכלל המשתמשים*
* *בכדי להשיג את אלו נדרשת גישה לקוד המקור (Source code)   
  של התוכנה*
* *אין צורך לשאול רשות או לשלם תמורת הפצה*
* *ההפצה היא חינמית או בתשלום*

### הרצת לינוקס

אחד הדברים היחודיים בלינוקס הוא הגמישות שלה. ניתן להריץ את לינוקס על מגוון פלטפורמות. את המערכת בגירסה רגילה, ניתן להריץ מ i386 ומעלה, על הגרסאות השונות של המחשב האישי.

למעשה, ישנן גם גרסאות למק (מעבדי פאוור פיסי ppc), גרסאות למעבדי אלפא, גרסאות למיינפרמיים, ומה שמעניין לא פחות, גם גרסאות שונות שרצות על מחשבי כף יד ואפילו שעונים מיוחדים. הפצות מודרניות של לינוקס, ידרשו מעבד חדיש יותר מ i386. כמו כן, להרצת מערכת גרפית, יש צורך בכמות זיכרון RAM גבוהה יחסית, כשלמערכות גרפיות כמו KDE3 הפופולארית רצוי מעל ל 128 זיכרון, ומעבד חזק יחסית.

* **לינוקס חופשית:** ניתן להוריד את לינוקס כולה, בחינם, מהאינטרנט. אין צורך לשלם דמי-הרשמה, העדכונים בחינם, קוד המקור זמין בחינם (למקרה שרוצים לשנות את המערכת). לינוקס חופשית כמו דיבור. הרישיון הנפוץ בו הוא GNU Public License (או בראשי תיבות GPL). הרישיון מעניק את הזכות לשנות את לינוקס, ובסופו של דבר להפיץ מחדש את שינויי הגירסה. זאת, בתנאי שהקוד ישאר זמין לאחר ההפצה מחדש. בפועל, יש לכל משתמש את החופש לקחת לדוגמה את הקרנל, להוסיף תמיכה במאפיין כלשהו, ולמכור את הקוד החדש, כל עוד הלקוחות שלך יוכלו להעתיק את הקוד.
* **לינוקס ניידת לכל פלטפורמת חומרה:** מוכר הרוצה למכור סוג חדש של מחשב ואינו יודע איזה מערכת הפעלה (OS) המחשב החדש יריץ, יכול לקחת את הקרנל של לינוקס ולגרום שיעבוד על החומרה שלו. זאת, מאחר שתיעוד הקשור לפעולה זאת, זמין באופן חופשי.
* **לינוקס נוצרה בכדי לרוץ:** בדומה ליוניקס, מצפים ממערכת לינוקס שתרוץ ללא איתחול (rebooting), כל הזמן. זו הסיבה לכך שמשימות רבות מבוצעות בלילה או מתוכננות אוטומטית, ומאפשרות זמינות גבוהה במהלך היום, תוך שמירה על יציבות החומרה. נכס זה, מאפשר ללינוקס להתאים גם לסביבות בהן למשתמשים אין אפשרות לשלוט על המערכת יום וליל.
* **לינוקס כמערכת הפעלה ורוב התוכנות שלה בעלות זמן תיקון קצר:** לינוקס מפותחת ונבחנת על ידי אלפי אנשים. כתוצאה כך, איתור תקלות, ומשתמשים שיתקנו תקלות, נמצאים בקלות. לעיתים קרובות, רק מספר שעות מפרידות בין מציאת באג והתיקון שלו.
* **לינוקס מאובטחת ורבגונית :**מודל האבטחה בלינוקס, מבוסס על אבטחת יוניקס, אשר ידועה כחסונה ובעלת איכות מוכחת. לינוקס, עומדת כמבצר נגד התקפות מהאינטרנט, באותה מידה גם נגד התקפות שונות, בסיוע הסטנדרטים הגבוהים של האבטחה. מכונת הפיתוח או תחנת הפיקוח יהיו מאובטחות בדיוק כמו "חומת האש" (Firewall).
* **לינוקס** **משתנה בהתאם לצרכים:** החל ממכשיר כף יד בעל 2MB זיכרון ועד למערכת אחסון גדולה, הכוללת מאות שרתים. ניתן להוסיף את החבילות הנחוצות או להסיר חבילות, ולינוקס תתאים לכל. לא צריך יותר מחשב-על, מכיוון שניתן להשתמש ביכולות המובנות של לינוקס, לביצוע דברים גדולים. בנוסף, ניתן ליצור מערכת הפעלה למערכת משובצת (כמו המחשב שיש במכונית) או רק להפעיל מחדש את המחשב 486 הישן.

לינוקס מפותחת תחת רישיון GNU General Public License, וקוד המקור שלה חופשי וזמין לכל אחד.

לינוקס והפצות לינוקס למינהן, אינן חייבות להיות חינמיות. חברות ומפתחים רשאים לגבות בעבור התוכנה כסף, כל עוד קוד המקור זמין עם התוכנה.

גרעין הלינוקס, הקרנל, יהיה זמין תמיד להורדה בחינם מהאינטרנט. רוב ההפצות, הקיימות כיום בשוק, גם הן ניתנות להורדה בחינם מהאינטרנט או לקניה במחיר סימלי.

#### הפצה (ראד האט, מנדרייק וכו')

למעשה, "לינוקס" היא רק הליבה (Kernel) של מערכת ההפעלה. רכיבים רבים דרושים על מנת לתפעל מחשב-אישי. לדוגמה, ספריות שונות, יישומים, קבצי אתחול וקבצי תצורה.

**הפצה** - אוספת את כל הנדרש מעבר לליבת המערכת, על מנת ליצור מערכת הפעלה שלמה ומלאה. היא גם "מפיצה" את האוסף הזה, בין אם על-גבי דיסקים (בד"כ בליווי ספרות ותמיכה), ובין אם בעזרת האינטרנט, להורדת קבצי ISO, לצריבה עצמית. כמו כן, לכל הפצה יש את מערכת ההתקנה הייחודית לה.

כל הפצה פונה לנישה אחרת של משתמשי מחשב, ולכן קיים מגוון הרחב. ישנן הפצות שמיועדות לשרתים, יש כאלה שמיועדות למחשבים חלשים, יש כאלה שמיועדות למחשבים מרובי מעבדים או כאלה שמטרתם העיקרית היא שוק המחשבים השולחניים.

**תמיכה** - אם ההפצה היא של חברה מאורגנת ומוכרת, יהיה קל לקבל תמיכה מהחברה עצמה או מאתר הבית שלה. הפצות פופולריות יקלו בקבלת תמיכה ברשת (מנועי חיפוש, פורמים..), מכיוון שליותר אנשים ישנה הכרות כלשהי עם ההפצה.

**תמיכה בחומרה** – הכרחית. הפצות רבות מחזיקות באתר שלהן, רשימת חומרה הנתמכת על ידן. רצוי לעבור על רשימה זו, כדי לבדוק אם החומרה שבמחשב נתמכת על ידי ההפצה. מומלץ לעבור על רשימה זו גם לפני שקונים חומרה חדשה, בכדי למנוע בעיות תמיכה.

**קלות שימוש והתקנה** - לא חיוני תמיד בשרתים, אבל למערכות ביתיות הוא כמעט הכרחי. הפצות שמכוונות אל שוק המערכות הביתיות, בדרך כלל, יהיו מצויידות באשף התקנות גראפי וקל לשימוש, ובכלים פשוטים להגדרת המערכת לאחר ההתקנה.

בחירת הפצה נעשית לפי שיקולים אישיים והמרכיבים שהובאו פה הם רק קווים מנחים שיסייעו במשימה זו.

**רד האט-**[http://www.redhat.com] נחשבת להפצה הפופולארית ביותר. ידועה מזה כשנים כנוחה יחסית למתחילים. מתאימה בהחלט גם לשימושים מקצועיים (שרתים וכו'). מאוד מומלצת לדעת רבים. כיום עולה כסף – גירסה חינמית שלה היא Centos.

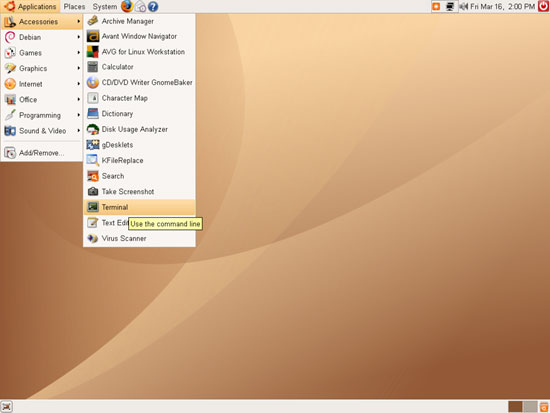
**אוּבּוּנטוּ (Ubuntu) )** מיועדת בעיקר לשימוש במחשבים אישיים, אך גם לטלפונים חכמים‏ ולמחשבי לוח. הפצה זו מתבססת על הפצת דביאן. אובונטו מתרכזת בעיקר בשימושיות, יציאה תקופתית של גרסאות חדשות, תהליך התקנה נוח וחופש מהגבלות חוקיות. אובונטו ממומנת על ידי קנוניקל, חברה פרטית שנוסדה על ידי היזם הדרום אפריקני מרק שאטלוורת'.

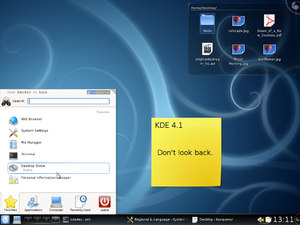
שם ההפצה מגיע מעקרון האובונטו של שבטי הזולו והקסוסה, אשר ניתן לסכם בפשטנות כ־"תפיסת עולם של שיתוף וחיבור בין כלל האנושות". הסיסמה של אובונטו – "לינוקס עבור בני אדם", מגדירה את אחת ממטרותיה - להפוך את לינוקס לזמין וקל לשימוש.

**Gnome או KDE?**

הפצה רגילה כוללת את ה-kernel ועוד הרבה כלים ויישומים. למשתמשי Windows, שחייהם סבבו סביב ממשק משתמש גרפי, החלק החשוב ביותר בהפצה הוא הכללת ה-X Windows System, שלרוב מתייחסים אליה בשם X. מערכת זו מאפשרת למשתמשי לינוקס להשתמש בסביבה גרפית ידידותית ומזמינה. יש שתי סביבות גרפיות עיקריות, שמשתמשות ב-X: KDE  K Desktop Environment), GNOME (ה-GNU Network Object Model Environment).

אחת הדרכים באמצעותן ניתן לקטלג את ההפצות השונות, היא בהתאם לשאלה אם הן משתמשות ב-KDE או ב-GNOME, כסביבה הראשונית שלהן. אין חובה לבחור דווקא באחת מהן. ניתן להתקין את שתיהן (וגם אחרות) בכל מערכת לינוקס. יישומים רבים, שפותחו בעבור סביבה אחת, ירוצו, לעיתים עם קצת יותר הורדות, גם בסביבה השנייה.

למרות שאפשר להתאים את שתי הסביבות, סביבת המחשב הראשונית של KDE עמוסה בצלמיות ומציגה למשתמש תפריטים רבים בגישה מיידית, בעוד שסביבת GNOME דוגלת בגישה נקייה יותר ופחות מתערבת.

הפצת Ubuntu משתמשת בגרסת ה-GNOME, אבל ניתן להוריד אתKubuntu  שכוללת, כפי שמרמזת האות הראשונה בשמה, ממשק ברוח KDE.

**מבנה מערכות הפעלה Unix ו- Linux**

מבנה מערכת ההפעלה: ליבה (Kernel) ומספר מערכות עזר (Utilities).

תפקיד הליבה כפול:

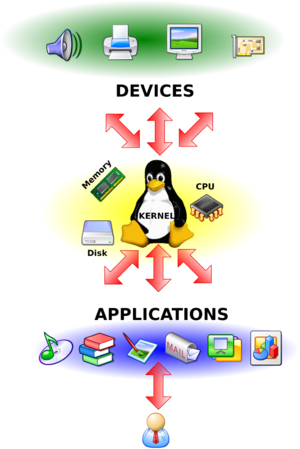
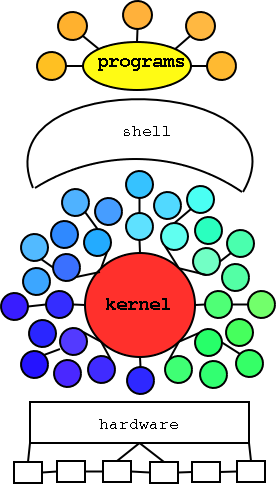
* לתווך בין החומרה לתכניות והמשתמשים
* לחלק משאבים בין משתמשים

הליבה מסתירה את פרטי החומרה ומאפשרת לתכניות ולמשתמשים להעזר בחומרה מבלי לדעת את פרטי היצרן או המבנה המדויק של החומרה.

הליבה נטענת לזיכרון בזמן ההפעלה ונשארת בזיכרון כל עוד המחשב דולק.

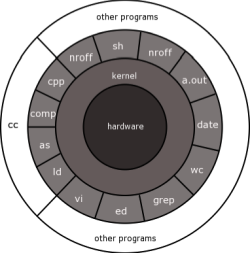
יתר הזיכרון מחולק בין התכניות והמשתמשים.

ביוניקס ולינוקס הליבה תופסת מעט זיכרון כך שלכל היתר נשאר יותר זיכרון.



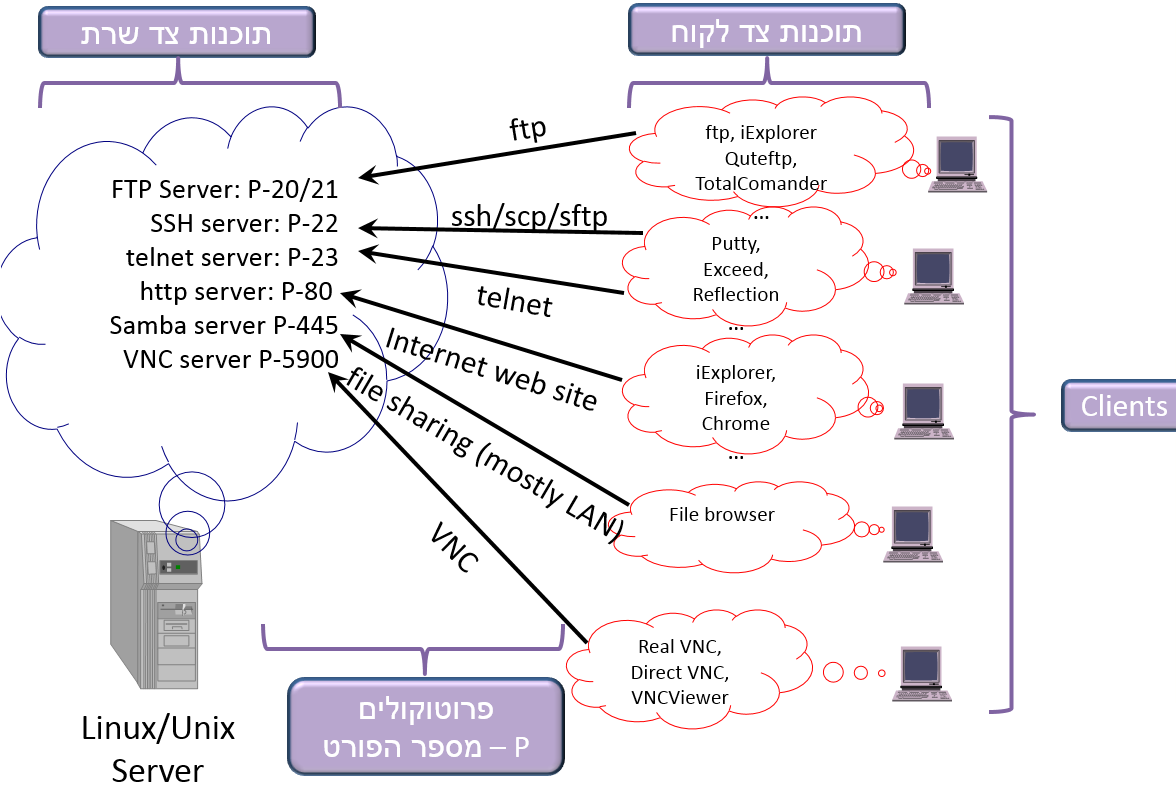
החלק הגדול של מערכת ההפעלה הן תכניות העזר יש מעל ל- 500 תכניות עזר במערכת

תכניות העזר מאפשרות:

* לנהל ולערוך קבצים
* לבדוק את משאבי המערכת
* לפתח תוכנות
* לנהל משתמשים
* לתקשר עם מערכות אחרות
* ועוד...

אחת מתוכנות העזר היא ה- Shell. תכנית זו מאפשרת לתת פקודות ברמת השורה (Interpreter) ואין לה זכויות מיוחדות - Interpreter היא תכנית הקוראת כל פעם שורה ומתרגמת אותה להוראות למחשב

### תקשורת שרת לקוח ב- Unix ו- Linux



**VNC**

בהתחלה היו מחשבים גדולים וחזקים שאליהם היו מחוברים מסופים.

מסוף כלל מקלדת ומסך (לפני היות העכבר) ו...זהו. זה היה צד לקוח רזה במיוחד– Thin Client. כל העבודה נעשתה על גבי השרת.

עם התפתחות המחשבים האישיים, יותר ויותר מהאחריות על ביצוע התכניות, עברה מהשרת ללקוח. בשיאם, דיברו על צד לקוח עשיר– Rich Client, שעשה חלק גדול מהעבודה בעצמו, בלי להזדקק לשרת כמעט. כשהאינטרנט נכנס והתקשורת הופכת מהירה יותר, מחשבי צד לקוח חוזרים להיות יותר ויותר רזים ומשאירים חלקים גדולים מהעבודה לצד השרת.

VNC – Virtual Network Computing – היא דוגמה ללקוח רזה מאד. על צד הלקוח מותקנת תוכנה קטנה שיוזמת התקשרות עם השרת (viewer) וכל יתר העבודה מתבצעת על השרת.

* צד הלקוח שולח הקלקות על עכבר והקלדות על המקלדת.
* צד השרת מבצע הקלדות והקלקות בשרת ושולח קטעים מתמונת המסך שהשתנו.
* צד הלקוח מעדכן את תמונת המסך בהתאם להוראות השרת.
* מאפשר התקשרות מכל מערכת הפעלה לכל מערכת הפעלה, וקיים עבור מרבית מערכות ההפעלה.
* מאפשר עבודה מרחוק ושימושי במיוחד בתמיכה טכנית ושירות לקוחות.

כיום ישנן תוכנות חינמיות רבות גם למערכת הפעלה חלונות העובדות עם פרוטוקול VNC.

VNC אינו תלוי בפלטפורמה מסוימת: לקוח VNC בכל מערכת הפעלה יכול להתחבר לשרת VNC בכל מערכת הפעלה אחרת. ישנם לקוחות ושרתים עבור כמעט כל מערכות ההפעלה, ואף עבור שפת התכנות Java. מספר לקוחות יכול להתחבר לאותו שרת VNC בו-בזמן.

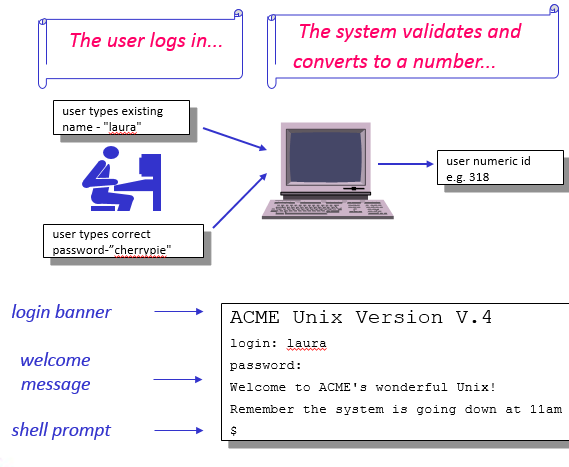
שימושים נפוצים עבור טכנולוגיה זו, כוללים תמיכה טכנית מרחוק, וגישה לקבצים מהמחשבים שבמקום העבודה דרך המחשב הביתי.

VNC פותח במקור על ידי AT&T. קוד המקור המקורי של VNC הוא קוד פתוח ברישיון GPL, כמו גם, רבות מתוכנות ה VNC הקיימות כיום.

למרות היתרונות, VNC סובל מחסרון עיקרי - אבטחה. הפרוטוקול עליו מתבססת התוכנה אינו מאובטח ומוצפן. לכן, המערכת עלולה לשמש לפריצה מרחוק למחשבים, אלא אם משתמשים בגירסה מסחרית מאובטחת, או מריצים את תוכנת השרת, על גבי חיבור מאובטח, כגון VPN.

|  |  |
| --- | --- |
|  | 1. **הורידו את התוכנה REAL VNC למערכת הפעלה חלונות** 2. **בצעו באמצעות התוכנה השתלטות על מחשב של חברכם** |

### כניסה למערכת וזיהוי משתמש

* הסיסמה לא נכתבת - גם לא בכוכביות או ברווחים.
* לאחר הכניסה מוצגת הודעה יומית.
* ה- Prompt משמעותו כי הופעלה תכנית ה- Interpreter והיא מחכה לפקודת המשתמש.

בסיום העבודה תמיד יש לצאת מהמערכת (לסיים את ה- Session) בצורה מסודרת ולא רק לכבות את המחשב.

יציאה מהמערכת משחררת משאבים, דואגת לשמור על הנתונים שלמים וחיונית לאבטחת מידע.

יציאה ממערכת Unix או Linux מתבצעת באמצעות פקודת **Exit** או הקשה על הלחצנים  
 Ctrl +d (**^d**).

### SHELL

SHELL - ממשק משתמש טקסטואלי, אשר מקבל פקודות מהמשתמש ומיישם אותן.

(MS-DOS prompt מזכיר shell באופן עבודתו במערכות Windows)

ה - shell מפרש את פקודות המשתמש ומיישם אותן, בהתאם לכללי תחביר (sysntex) אופיינים.

בלינוקס ישנם מספר סוגי Shell, אשר הינם בעלי אופי ייעודי שונה. הנפוצים ביותר בשימוש מכונים BASH ו-SH, וההבדלים ביניהם מעטים.

על מנת שמשתמש יוכל לעבוד ב Shell, יש להגדיר עבורו את סוג ה Shell בקובץ passwd. סוג זה ישמש כברירת מחדל.

ניתן לעבור לסוג Shell אחר במהלך העבודה וכן ניתן להגדיר קיצורים לפקודות להקלת העבודה.

ישנם סוגי Shell שונים, הנפוצים הם:

* Bourne shell **sh**
* Bourne again shell **Bash**
* C shell **csh**
* Korn shell **ksh**

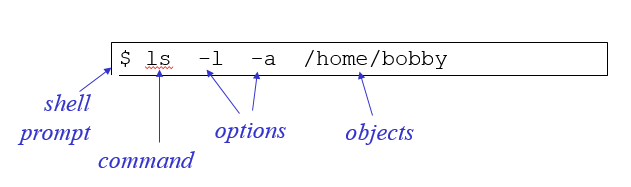
כיוון שה- Shell הוא תכנית, ככל תכנית אחרת, ולא חלק מליבת מערכת ההפעלה, אפשר להחליף בין סוגי ה- Shell השונים. ה-Shell הנפוץ ביותר כיום הוא ה-ksh, שמאפשר גישה להיסטוריית הפקודות ועריכת שורת הפקודה.

#### טרמינל (Terminal)

טרמינל הינה תוכנה, אשר מאפשרת התחברות למחשב מרוחק (או המחשב המקומי) ולפתוח Shell. קיימים סוגי טרמינלים רבים מסוגים שונים, המאפשרים שליטה על ממשק המשתמש כגון פונט, גודל צבע וכו...

### הכנסת פקודות באמצעות ה- Shell

|  |  |
| --- | --- |
| Case Sensitive | * הקבצים a ו- A הם שני קבצים שונים |
| פקודה היא הדבר הראשון בשורה | * יתר השורה מכילה ארגומנטים משני סוגים:   1. אלו המתחילים לרוב בסימן "–" : האופציות המשנות את הדרך בה הפקודה פועלת. ניתן לאחד "-" בודד לשלב מספר אופציות, לדוגמה: -la שקול ל: -l -a   2. אלו המתחילים ללא סימן מקדים, מצביעים על האובייקט שעליו הפקודה פועלת (לרוב שם קובץ או ספריה) * גם לאופציות יכולים להיות ארגומנטים * לכל פקודה יש רשימת אופציות ייחודית לה |
| פקודות ופרמטרים מופרדים על ידי "רווח לבן" | * "רווח לבן" יכול להיות רווח או TAB |
| סוף פקודה נקבע על ידי הקשת <CR> | * <CR> הוא המקש המוכר כ- Enter או כ- Return * CR הוא קיצור של Carriage Return, מושג שקשור במקור למכונות הדפסה ידניות |
| Prompt ברירת המחדל | * $ for the Bourne and Korn shells * % for the C shell * # for any shell run by the super user   קל לשנות את ה- Prompt כך שיכיל מידע מועיל (כמו למשל המיקום בהיררכיית הקבצים, או השעה והתאריך) |

**מבנה הפקודה**

* פקודות נכתבות אחרי ה- Prompt
* בין החלקים השונים מפרידים רווחים

פקודות בUNIX- הן למעשה תכניות, ולכן הקשת שם פקודה היא בקשה למערכת ההפעלה להריץ תכנית מסוימת, המאוחסנת במקום מסוים. למעשה, שם הפקודה מהווה שם תכנית. התחביר (SYNTAX) של הפקודות הוא כדלהלן:

**<רשימת פרמטרים> <שם פקודה/תכנית><enter >**

דוגמאות:

* date
* cp f1 f2
* ls -l d1 d2

הרכיבים ברשימת הפרמטרים הם מחייבים או אופציונליים:

* **רכיב מחייב** משדר למערכת ההפעלה "מה לבצע".
* **רכיב אוֹפציונלי** משדר למערכת ההפעלה "איך לבצע" את הפקודה.

בדוגמה לעיל, הפרמטרים d1 d2 מורים לפקודה ls להציג את הקבצים d2 ו- d1, והפרמטר -l מורה להציג את התוצאות בפורמט מוּרחב.

ב-UNIX מקובל להתחיל פרמטר אופציונלי בתו "-".

עם סיום מתן הפקודה, יש ללחוץ "enter", ובכך ניתנת למערכת ההפעלה הודעה לבצע את הפקודה,

ניתן להפסיק פקודה במהלך ביצועה או לא לבצע פקודה, ע"י **Ctrl+C**.

### תווים מיוחדים



### פקודות לקבלת עזרה

* Whatis *command or filename* – תיאור קצר
* man *command* - עזרה מלאה לגבי פקודה
* man –k *keyword*– עזרה מלאה לגבי נושא – התוצאה תהיה רשימת הפקודות / קבצים הרלוונטיים כאשר לכל פקודה מוצמד מספר

דפי עזרה של Unix מתחלקים לשמונה פרקים המספרים מייצגים סעיף בדפי העזרה:

1. פקודות משתמש ברמת מעבד הפקודות

2. תכניות מערכת (system calls) מתוך ספריות C

3. פונקציות נוספות מתוך ספריות C

“special files” ודרייברים התקנים .4

5. תיאור סוגי קבצים

6. משחקים

7. שונות

8. תכניות לתחזוק המערכת ומידע

דוגמאות:

**man 1 mkdir**

**man mkdir**

**man 2 mkdir**

**=> There is a system call mkdir!**

**man grep**

**man 5 passwd**

### פקודות פשוטות

* date [+format] – הצגת תאריך ושעה נוכחיים
* cal [[month] year] – הצגת לוח שנה
* w or who – הצגת משתמשים מחוברים
* echo [arguments] – הדהד על המסך

|  |  |
| --- | --- |
|  | 1. כנסו למערכת 2. השתמשו בעזרה כדי להשלים את המשימות הבאות: 3. הדהדו על המסך את המשפט "hello world" 4. הציגו את התאריך והשעה הנוכחיים 5. הציגו לוח שנה לחודש לשנת 2000, באיזה יום בשבוע היה הראשון לינואר? 6. הציגו לוח שנה לחודש ספטמבר שנת 1752 7. מה ההבדל בין פקודת w לפקודת who? 8. הציגו את התאריך הנוכחי בפורמט mm/dd/yy 9. הציגו את שם החודש המלא (*לדוגמה – January*) 10. הציגו את התאריך שהיה לפני שלושה ימים 11. הציגו רק את השעה, באיזה פורמט השתמשתם? 12. הציגו את יום השבוע, לתאריך הנוכחי |

#### טבלת השוואה בין פקודות בסיסיות בין שתי מערכת ההפעלה

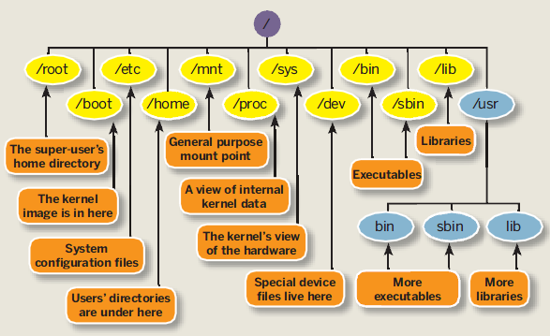
|  |  |  |
| --- | --- | --- |
| **ייעוד** | **פקודה בחלונות** | **פקודה בלינוקס** |
| רשימת קבצים בתיקיה | dir | is |
| שינוי תיקיה | cd | cd |
| הצגת path מלא של תיקיה נוכחית | chdir | pwd |
| ניקוי המסך | cls | clear |
| יצירת תיקיה | mkdir | mkdir |
| העתקת קובץ | copy | cp |
| מחיקת קובץ | del | rm |
| הזזת קובץ | move | mv |
| שינוי שם קובץ | ren | mv |
| מחיקת תיקיה | rmdir | rm –rf |
| צפייה בקובץ ("הזרמה" שלו לתוך ה shell). שימושי יותר בלינוקס כאשר ממשק גראפי לא זמין | type | cat |
| צפיה בקובץ עם גלילה | TYPE <filename> | more | cat <filename> | less |
| עזרה על פקודה מסוימת | <command> /? | man <command> (עם ניווט) |

### Unix ו- Linux מערכת הקבצים

מידע ותכניות נשמרים בקבצים. הקבצים מאורגנים באמצעות ספריות (Directories), למבנה דמוי "עץ" היררכי. ספריה היא למעשה קובץ מיוחד שיש בו מידע על קבצים אחרים ועל ספריות אחרות.

ב- Unix כל דבר הוא קובץ. אפילו לחומרה ניגשים דרך קבצים מיוחדים. בגרסאות החדשות של Unix, ניגשים דרך מערכת הקבצים גם לזיכרון ולתכניות שרצות במחשב.

מבנה מערכת הקבצים הוא היררכי. בכל עומק יכולים להיות קבצים וספריות ללא הגבלת מספר. תמיד אפשר להוסיף עוד קובץ או ספריה לענף. למבנה עץ הקבצים יש שורש יחיד (Root), שממנו אפשר להגיע לכל הקבצים, הספריות והדיסקים במערכת, ואפילו להתרחב למערכות אחרות.

* המיקום בעץ הקבצים אינו תלוי דיסק – כל הדיסקים מאוגדים לעץ קבצים אחד עם שורש (**Root**) בודד.
* המיקום הנוכחי בעץ נקרא Current או **Working directory**.
* לכל משתמש במערכת יש **home directory** משלו, שבו יש לו הרשאות מלאות והוא גם המיקום הנוכחי לאחר הכניסה למערכת.
* לכל משתמש ב- home directory יש קבצים מיוחדים, המופעלים בתחילת ה- Session. קבצים אלו מתחילים לרוב בנקודה.
* קבצים דוגמת .profile, .login, .bashrc משמשים להתאמת פעולת המערכת עצמה (*למשל, איך יראה ה- prompt של ה- shell*).

|  |  |
| --- | --- |
| **אבסולוטי Absolute** | **יחסי Relative** |
| מתחיל תמיד משורש העץ root | מתחיל מהמיקום הנוכחי בעץ  Working directory |
| מתחיל בסימון / | מתחיל ללא סימון / |
| כולל את המסלול המלא מהשורש אל הקובץ או הספריה הנבחרים | כולל את הניווט הדרוש בעץ מהמיקום הנוכחי אל הקובץ או הספריה הנבחרים |
| /usr/bin/tty | ../bin/tty |
|  | . – מייצג את המיקום הנוכחי .. – מייצג את ספריית ההורה |

### שמות קבצים ו- Wildcards

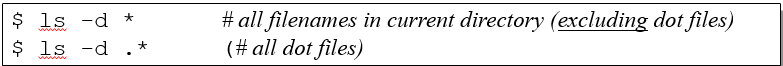
* שם קובץ יכול להכיל כל תו חוץ מ- / .
* תווים רבים יכולים להתגלות כבעייתיים לכן מקובל להשתמש בתוים אלפאנומריים (אותיות ומספרים), נקודה ומקף בלבד.
* שמות קבצים אינם מכילים סיומות. שימוש בסיומות הוא סטנדרט לא הכרח.
* קבצים ששמותיהם מתחילים בנקודה הם קבצים מיוחדים שאינם מוצגים בדרך כלל.
* בגירסאות ישנות שם קובץ היה מוגבל ל-14 תווים, בגרסאות חדשות ההגבלה היא ל- 255 תווים.
* בפקודות מסוימות (למשל ls) אפשר להחליף חלק מהתוים בשם הקובץ בסימנים מיוחדים:

\* - סדרת תווים כלשהי (יכולה גם להיות באורך אפס)

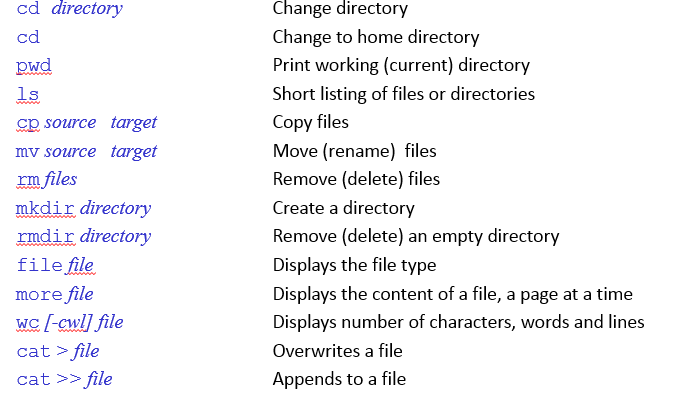
? - תו בודד כלשהו

[*list*]- תו בודד שנכלל ברשימה שבין [ ]

[!*list*] - תו בודד שלא נכלל ברשימה שבין [ ]



### פקודות פשוטות על מערכת הקבצים



אחת הפקודות השימושיות ביותר היא פקודת ls. פקודה זו מציגה רשימת קבצים וספריות.

הוספת אפשרויות (Options) לפקודה גורמת לשינוי ברשימה המוצגת. האפשרויות הנפוצות: -a, -d, -l, -R

* ls –a מציגה גם את הקבצים המוסתרים
* ls –d מציגה את אפשרויות הספריה במקום התכנים
* ls –l מציגה מידע נוסף הקשור לקבצים (גודל, הרשאות וכדומה)
* ls –R מציגה את הספריה המבוקשת וכל הספריות תחתיה

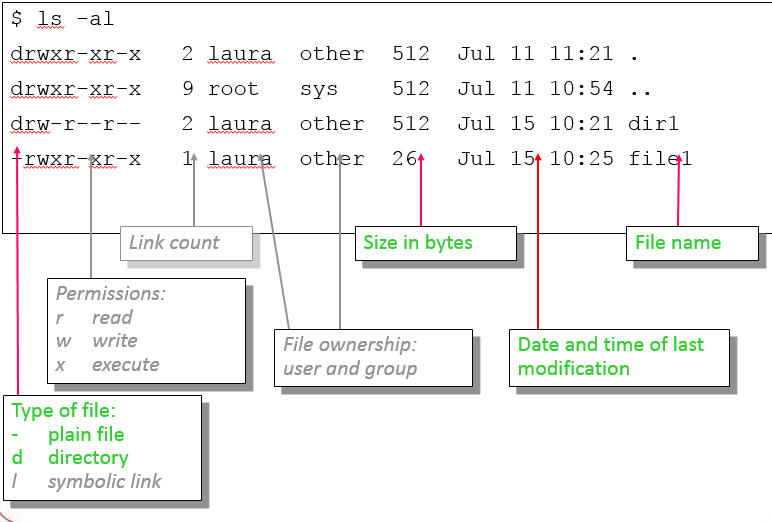
*לדוגמה*:

* *$* ***ls -l /tmp****total 211  
  -rw-r--r-- 1 root root 9701 Mar 21 12:27 install.log  
  drwxr-xr-x 2 root root 12288 Mar 21 03:08 lost+found  
  drwx------ 2 alina alina 1024 Mar 23 11:35 myhome  
  -rw------- 1 alina alina 118784 Mar 21 23:21 mypass*

*לעומת*

* ***$ ls -ld /tmp****drwxrwxrwt 5 root root 1024 Mar 23 13:01 /tmp*

**פקודת ls בתצוגה מורחבת**



לקבצים יש צבעים. צבע הטקסט הוא בעל משמעות לפי סוג הקובץ:

* כחול – תיקיה
* ירוק – קובץ הניתן להרצה (executable)
* אדום – קובץ דחוס (rpm, tar, zip)
* וכו'

ישנם צבעים המסמנים תכונות:

* רקע ירוק עם טקסט שחור (כמו tmp) אומר שזו תיקיה עם הרשאות כתיבה ל"כולם"'

#### העתקת קבצים, פקודת cp

* הפקודה cp משמשת להעתקת קבצים או ספריות
  + cp [options] source target
  + cp [options] source … directory
* מספר אפשרויות מועילות:
  + -i interactive checking (prompts only if target exists)
  + -r copy directories recursively
* אם מוגדר יותר מקובץ מקור בודד, המטרה חייבת להיות ספריה.

$ cp /etc/passwd passwd

$ cp -i /etc/p\* .

$ cp -r . /tmp/myhome

* אם מוגדר קובץ בודד, המטרה היא שם העותק שיווצר (קובץ).
* כדי להעתיק קבצים, עליכם להיות בעלי הרשאות כתיבה בספריית היעד.

#### העברת ושינוי שם, פקודת mv

* השתמשו בפקודת mv כדי לשנות שם קובץ או להעתיק קבצים/
  + mv [options] source target
  + mv [options] source … directory
* אפשרויות שימושיות:
  + -i interactive checking (prompts only if target exists)
* אם מוגדר יותר מקובץ מקור בודד, המטרה חייבת להיות ספריה.

$ mv passwd /tmp/mypass

$ mv -i file1 /tmp/mypass

mv: replace ‘/tmp/mypass’?

* אם מוגדר קובץ בודד, והמטרה היא ספריה – הקובץ יועבר אל הספריה.
* אם מוגדר שם קובץ בודד והמטרה היא שם קובץ (ללא ציון ספריה) שם הקובץ ישתנה.
* אם מוגדר שם קובץ בודד והמטרה היא שם קובץ בספריה מוגדרת – שם הקובץ ישתנה והקובץ יועבר לספריה המוגדרת.
* אם קובץ המקור הוא ספריה, אז שם הספריה ישתנה (לא ניתן להעביר ספריה/).
* אם קובץ המטרה קיים – הוא ידרס Overwrite.
* קובץ יכול להדרס אם יש הרשאות כתיבה לספריה בה הוא נמצא, גם אם אין הרשאות כתיבה לקובץ עצמו.

#### מחיקת קבצים, פקודת rm

* השתמשו בפקודת rm כדי למחוק קבצים
  + rm [-ir] file1 [file2...]
* אפשרויות שימושיות:
  + -i interactive checking (prompts for each file)
  + -r recursive descent of directories
* אפשר למחוק קובץ אם יש הרשאות מחיקה לספריה בה הוא נמצא.

$ rm mypass

$ rm -i my\*

$ rm -r /tmp/myhome

* אם אין הרשאות מחיקה לקובץ עצמו אבל יש הרשאות מחיקה לספריה, ניתן יהיה למחוק אחרי אישור.
* שימוש באפשרות –r מאפשרת למחוק קבצים בספריה הנוכחית וכל הספריות תחתיה.
* אי אפשר להשתמש בפקודה rm כדי למחוק ספריה, גם אם היא ריקה. ניתן להשתמש בפקודה rmdir.

#### יצירת ומחיקת ספריות ריקות, פקודות *mkdir* ו-*rmdir*

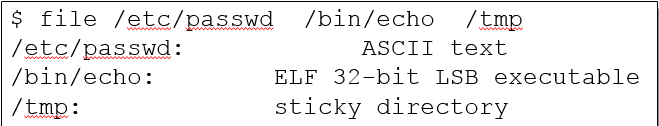
* פקודת mkdir משמשת ליצירת ספריה ריקה.
* פקודת rmdir משמשת למחיקת ספריה ריקה.
  + mkdir directory
  + rmdir directory
* כדי למחוק ספריה המכילה קבצים, ראשית יש למחוק את הקבצים, ואחר כך למחוק את הספריה הריקה.

#### מידע על קבצים – פקודות *file*, *wc*

פקודות שונות מאפשרות לקבל מידע על הקובץ או להציג את תוכנו.

* הפקודה **file** מציגה מהו סוג הקובץ

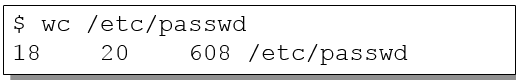
file [*file...*] Determines the content type of file (text, executable, C code,directory, etc)



* הפקודה **wc** מציגה את מספר התווים, המילים והשורות שהקובץ מכיל

wc[-cwl][*file…*] Counts characters, words and lines in a file

שימוש באפשרויות הפקודה מגביל את התצוגה לתווים, מילים או שורות בלבד.



#### הצגת תוכן קבצים, הפקודה more

* השתמשו בפקודה more כדי להציג את תוכן הקובץ
* אם הקובץ מכיל יותר מעמוד אחד על המסך, הקובץ יוצג עמוד אחרי עמוד

more [*file...*] Displays contents of a file, one page at a time

* לאחר הצגת קובץ אחד, הפקודה תחכה להוראות כדי לדעת איך להמשיך.

הפקודות האפשריות הן:

*<space>* next page

*<CR>* next line

*b* back one page

*f*  skip forward one page

*q*  quit

*/string* search for specified *string*

*h* display help page

* בכל מערכות Unix הפקודה more קיימת.
* ברוב מערכות Unix המסחריות גם פקודת pg קיימת (עושה אותו הדבר, אבל הפקודות להעברת דפים שונות).
* במערכות Linux יש לרוב את הפקודות more ו- Less.

#### הצגת תוכן קבצים, הפקודות *pg* ו- *less*

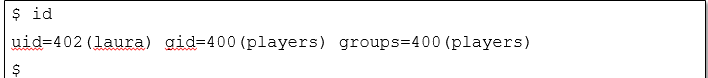
|  |  |  |
| --- | --- | --- |
| **הפקודה** | **Pg** | **Less** |
| עזרה | h | h |
| הדף הבא | <CR> | <Space> |
| השורה הבאה | <Space> | <CR> |
| הדף הקודם | - | b |
| חיפוש תבנית | /pattern | /pattern |
| חצי הדף הבא | d | d |

* אפשר ליצור קבצים על ידי שימוש בפקודה cat או על ידי שימוש בעורך טקסט (Editor) כגון: vi, emacs, nedit, nano, pico.
* בפקודה cat משתמשים לרוב בשילוב עם פקודה אחרת באמצעות pipeline ( | ).
* ls –la | cat >> aaa append to file aaa the file list created by the command *ls-la*0

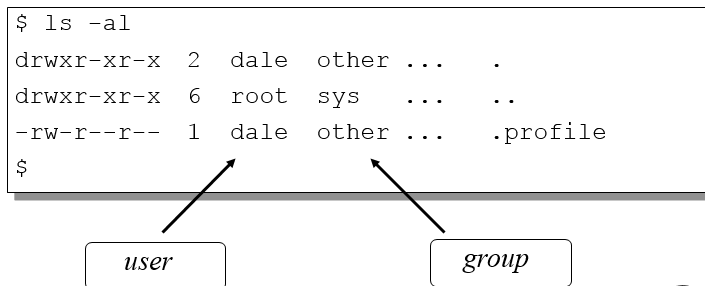
|  |  |
| --- | --- |
|  | 1. מה עושות הפקודות הבאות?   *$ ls –ld \*.?*  *$ more [A-Z]\**  *$ ls –ld /etc/[!a-m]\**  *$ file /usr/bin/\*x\**  *$ ls –ld [a-z]\*[0-9]*  *$ ls* *-a .[!.]\**   1. צרו את הספריה Data ובתוכה את הקבצים הבאים (שים לב לקבצים נסתרים): ,my\_name data1,data2,data\_3,.data. 2. צרו את הספריה Zoo ובתוכה את הקבצים : keeper,animals,Visitors 3. צרו את הספריה Progs ובתוכה את הקבצים: prog1.c,prog2.c,small\_prog,medium\_prog,.pro1.exe,my\_prog. 4. צרו את הספריה Family בתוך ספרית Zoo 5. השתמשו בעורך הטקסט **nano** כדיליצור קובץ בשם My\_name וכתבו בו את שמכם בשורה ראשונה וכתובתכם בשורה השנייה 6. צרו קובץ בשם My\_family וכתבו בו שמות שלושה בני משפחה – אחד בכל שורה 7. וודאו כי הקבצים תחת ספריית Family 8. העתיקו את קובץ My\_name לקובץ בשם name 9. שנו שמו של קובץ My\_family ל - Family 10. צרו קובץ בשם Whole\_family המורכב משרשור Family ו - name   **cat name Family >> Whole\_family**   1. הציגו את Whole\_family 2. עברו ל home directory- שלך 3. שנו את שם ספריית Zoo ל My\_zoo 4. העתיקו את ספריית Family על תוכנה ל - home directory שלך 5. מחקו את ספריית My\_zoo על תוכנה 6. חשבו את נפח ספריית הבית שלכם באמצעות הפקודה: **du –hs ~** |

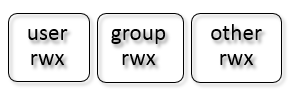
### מערכת ההרשאות וארגון הקבצים

* כל משתמש שייך לקבוצת ברירת מחדל המוגדרת ב-/etc/passwd.
* משתמש יכול להיות שייך גם לקבוצות אחרות המוגדרות ב-/etc/group.
* קבצים אלו מתוחזקים על ידי מנהל המערכת. בקובץ group מוגדרות הקבוצות הקיימות ומספרן (לרב ישקפו את מבנה הארגון או שם הפרויקט).
* הפקודה id מציגה את המשתמש הנוכחי והקבוצות אליהן הוא שייך.
* קובץ נוצר עם המשתמש והקבוצה הנוכחיים, לכל קובץ יש בעלים (Owner) אחד בלבד והוא שייך לקבוצה אחת בלבד רק בעל הקובץ יכול לשנות את הרשאות הקובץ.
* הרשאות ספריות פועלות בדיוק כמו הרשאות קבצים.

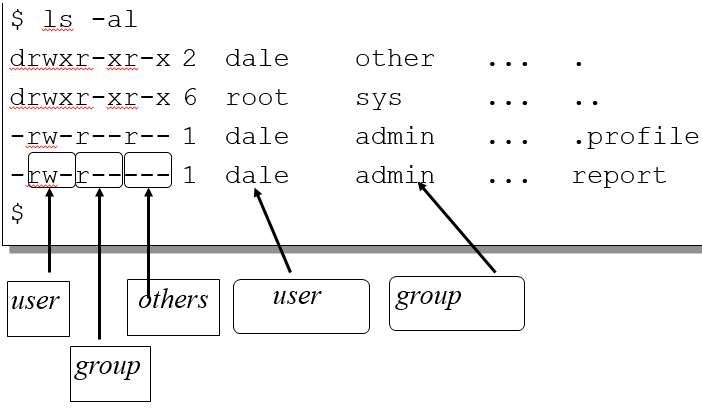


* בפקודת ls –la אפשר לראות לאיזה משתמש לאיזה קבוצה הקובץ שייך



* הפקודה cp יוצרת קובץ חדש, עם בעלים וקבוצה חדשים.
* הפקודה *mv* משאירה את הקובץ עם הבעלים והקבוצה המקוריים.
* לכל קובץ וספריה מוגדרות 3 רמות הרשאה:
* User - למשתמש עצמו
* Group - לקבוצה (כל המשתמשים בקבוצה)
* Other - ולכל האחרים
* אפשר להגדיר, בכל רמה, 3 סוגי הרשאות:
* קריאה Read
* כתיבה Write
* הפעלה Execute

למשתמש-על (Super user) יש הרשאה לעקוף את ההרשאות.

* ההרשאות מצוינות על ידי 9 דגלים המחולקים לשלוש קבוצות:
* למשתמש דייל יש הרשאות כתיבה וקריאה
* לקבוצה Admin יש הרשאות קריאה בלבד
* ליתר אין הרשאות על הקובץ כלל

הדגל הראשון מציין האם ההרשאות הן לקובץ (-) או לספריה (d)

#### איך קובעים את סוג ההרשאה למשתמש?

#### איזה דגלון מודיע על סוג ההרשאה?

READ **r**wx - *access contents (and copy)*

WRITE r**w**x - *update contents (not delete)*

EXECUTE rw**x** - *executable programs*

rw**x** - *shell scripts*

* מה ההרשאות ההפעלה בספריה /usr/bin ?
* בספריה הזו נמצאים כל תוכנות העזר של מערכת ההפעלה שלכל משתמש מותר להפעיל.
* מה הרשאות ההפעלה בספריה /usr/sbin ?
* בספריה הזו לרב המשתמשים הרגילים אין הרשאות הפעלה.

#### אילו הרשאות צריך כדי לבצע פעולות על קבצים וספריות?

LIST rwx *list directory contents, e.g. ls*

SEARCH rwx *must be able to search directory to locate and work on files*

CREATE FILE rwx

RENAME FILE rwx *file permissions are not relevant*

DELETE FILE rwx

* שם הקובץ הוא לא חלק מהקובץ אלא חלק מהספריה, לכן, כדי למחוק או לשנות שם יש צורך בהרשאות לספריה ולא לקובץ עצמו.
* עבור ספריות, הרשאת "הפעלה" משמעותה למעשה הצגה.
* לא ניתן להציג תוכן ספריה (או תוכן ספריות מתחתיה) ללא הרשאת הפעלה על הספריה.

#### איך משנים הרשאות?

* אפשר לשנות הרשאות בעזרת הפקודה **chmod *user* [=+-] *flag files...***

= sets permission

+ adds permission

- removes permission

* כאשר משתמש יכול להיות אחד מאלו: user u

g group

o other

a all types of users (collective option, same as ugo)

* ודגלון הגישה יכול להיות אחד מאלו: read r

w write

x execute

למשל:

**chmod +x** *add execute permission for all*

**chmod g+rw** *add read and write permission for group*

**chmod ug-w** *remove write permission for user and group*

**chmod og=rx** *set other and group access to read and execute (removing write permission if set)*

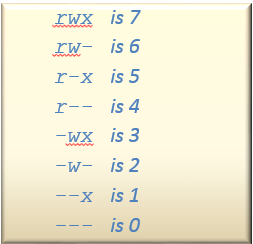
**chmod ug+x,o-rwx** *add execute permission for user and group. Remove all permissions for others*

### הרשאות (קיצורי דרך למתקדמים)

שיטה אחרת לשנות הרשאות היא באמצעות קוד אוקטלי.

**chmod number**

חשוב להזהר ולהכניס את המספר הנכון, כי אחרת יכולים לאבד הרשאות חשובות לקבצים ולספריות.



* המספר מורכב תמיד משלוש ספרות
* כל ספרה מייצגת קבוצת הרשאות
* הספרה הראשונה מייצגת את הרשאות המשתמש
* הספרה השנייה מייצגת את הרשאות הקבוצה
* הספרה השלישית מייצגת את ההרשאות של כל היתר

לחישוב כל ספרה פעלו בדרך הבאה:

* קריאה = 4
* כתיבה = 2
* הפעלה =1

לחישוב בספרה חברו את ההרשאות הדרושות .למשל, הרשאה 750 משמעותה

* הרשאות מלאות למשתמש
* הרשאות קריאה והפעלה לקבוצה
* אין הרשאות לאחרים

|  |  |
| --- | --- |
|  | 1. מה משמעות הפקודות הבאות?   $ chmod u+x *files...*  $ chmod a+r *files...*  $ chmod g+rw *files...*  $ chmod og-w *files...*  $ chmod -R go-wx *directory*  $ chmod og=rx *files...*  $ chmod go= *files...*   1. אם העתקנו קובץ שמישהו אחר יצר, מי הבעלים של העותק? 2. האם ניתן למחוק קובץ שיש לנו עליו הרשאות כתיבה וקריאה אם אין לנו הרשאות כלל על הספריה שבה הוא נמצא? 3. אם נתנו לכל הקבצים בספריה את ההרשאה 777, למי יש הרשאה לעשות מה? 4. אם נתנו לכל הקבצים בספריה את ההרשאה 700, למי יש הרשאה לעשות מה? 5. אם נתנו לכל הקבצים בספריה את ההרשאה 764, למי יש הרשאה לעשות מה? 6. אם נתנו לכל הקבצים בספרית הבית (home) את ההרשאה 000, באיזו בעיה אנו עלולים להתקל? |

**לסיכום: משפחת מערכות הפעלה Unix ו- Linux**