எளிய தமிழில்

பாகம் - 2

து. நித்யா

கணியம் வெளியீடு

http://kaniyam.com

எளிய தமிழில் GNU/Linux

பாகம் – 2

து. நித்யா

எளிய தமிழில் GNU/Linux பாகம் – 2 Copyright © 2014 by

Creative Commons Attribution-ShareAlike 3.0 Unported.

பொருளடக்கம்

- எளிய தமிழில் GNU/Linux பாகம் – 2
- ஆசிரியர் உரை
- சமர்ப்பணம்
- நூல் வெளியீடு
- Main Body
- Linux கெர்னலின்-ன் விரிவான வரலாறு
- Users-ஐ கையாளுதல்
- Disk Management
- Mount மற்றும் Unmount
- Process Management
- உபுண்டு லினக்சில் மென்பொருட்களை நிறுவுதல்

- Shell Programming பற்றிய அறிமுகம்
- Shell Script- 2
- Shell Scripting – 3
- Job Scheduling
- Rsync மூலம் backup செய்தல்

எளிய தமிழில்

பாகம் - 2

து. நித்யா

கணியம் வெளியீடு

http://kaniyam.com

ஆசிரியர் உரை

ஏதாவது புதிதாக செய்ய வேண்டும் என்ற எண்ணம் எனக்குள் எப்போதும் ஓடிக்கொண்டே இருக்கும். அதை நோக்கி நான் தொடங்கும் ஒரு சில முயற்சிகளை, அன்றாட வேலைப்பளுவின் காரணமாக பாதியிலேயே விட்டுவிடுவேன். பின்னர் எப்போதும் போல் என் வாழ்க்கை, "ஏதாவது புதிதாக செய்ய வேண்டும்" எனும் எண்ணத்தைத் தாங்கிக் கொண்டு சாதாரணமாக சென்று கொண்டிருக்கும்.

அப்போது தான் எனது பல முயற்சிகளில் ஒன்றாக, MySQL-க்கு தமிழில் ஒரு புத்தகத்தை எழுதத் தொடங்கினேன். பின்னர், சில மாதங்களுக்கு முன்பு

“கணியம்” இதழ் மூலம் அந்தப் புத்தகம் வெளியிடப்பட்டது. அதற்குக் கிடைத்த மாபெரும் வரவேற்பு என்னை மிகவும் மகிழ்ச்சிக்கு உள்ளாக்கியது. இந்தப் புத்தகத்திற்கு இவ்வளவு பெரிய வரவேற்பும், பாராட்டுக்களும் என்னை வந்து சேரும் என்று நான் சற்றும் எதிர்பார்க்கவில்லை. உண்மையிலேயே நான் ஏதோ உருப்படியாக செய்திருக்கிறேன் என்று என் மேல் எனக்கே மதிப்பு வரத் தொடங்கியது.

இவ்வாறு நான் அடைந்த மகிழ்ச்சியே, என்னை மீண்டும் GNU/Linux-க்குத் தமிழில் ஒரு புத்தகம் எழுதுவதற்குத் தூண்டியது. எனது முயற்சிகளுக்கு ஊக்கமளிக்கும் வகையில், என்னைப் பாராட்டிவரும் ஒவ்வொருவரும் தான், நான் இந்தப் புத்தகம் எழுதுவதற்குக் காரணமானவர்கள்.

பாகம் 1 ஐ தொடர்ந்து, இதோ ‘எளிய தமிழில் GNU/Linux –

பாகம் - 2

கட்டற்ற கலைக்களஞ்சியமான தமிழ் விக்கிபீடியாவின் பத்தாண்டுகள் நிறைவு விழாவில், இந்த நூலை வெளியிடுவதில் பெரும்கிழ்ச்சி கொள்கிறேன்.

என்னை இந்தப் புத்தகம் எழுதுவதற்கு ஊக்கமளித்த அனைவருக்கும் நன்றி.

து. நித்யா

கிழக்கு தாம்பரம்,
சென்னை
29செப்டம்பர்
2013

மின்னஞ்சல்:
nithyadurai87@gmail.com

வலை பதிவு: http://nithyashrinivasan.wordpress.com

சமர்ப்பணம்

GNU/Linux க்கு உழைக்கும் அனைவருக்கும் இந்தப் புத்தகம் சமர்ப்பணம்.

நூல் வெளியீடு

செப்டம்பர் 30, 2013 அன்று சென்னையில் தமிழ் விக்கிபீடியாவின் பத்தாண்டுகள் கொண்டாட்ட நிகழ்ச்சியில், இந்த நூல் வெளியிடப்பட்டது.

முதல் பிரதியை வெளியிட்டவர் : திரு. சி.இரா.செல்வகுமார் அவர்கள், பேராசிரியர், வாட்டர்லூ பல்கலைக்கழகம், கனடா (வலது கோடி)

பெறுபவர் : திரு.சுந்தர் அவர்கள், 2004-ஆம் ஆண்டு முதல் ஆங்கிலம், தமிழ் விக்கிப்பீடியாக்களில் பங்களித்து வரும் தகவல்பெறுநுட்ப வல்லுனர். (இடது கோடி)

ஊக்கத் தொகை

தமிழில் நுட்பங்களை எழுதி, கிரியேட்டிவ் காமன்ஸ் எனும் கட்டற்ற உரிமத்தில் வெளியிடுவதை பாராட்டி,

செல்வா-குமரி அறக்கட்டளையின் பரிசாக, திரு. சி.இரா.செல்வகுமார் அவர்கள் 100 டாலர் (6500 ரூ) அளித்தார்.

இவர் ஏற்கெனவே, “எளிய தமிழில் MySQL” நூலுக்காக 5000 ரூ நன்கொடை அளித்துள்ளார்.

ஒளிப்படம்: http://jasanpictures.com

Creative Commons உரிமையில் தமது நூல்களை

வெளியிடும் எழுத்தாளரை உங்கள் நன்கொடைகள்

ஊக்குவிக்கும்.

வங்கி விவரங்கள்.

Name – Nithya Duraisamy

ICICI – 006101540799

Branch – Mcity branch, chengalpattu.

IFSC code – ICIC0000061

Linux கெர்னலின்-ன் விரிவான வரலாறு

பொதுவாக வரலாறு என்பது படிப்பதற்குச் சற்று கடினமாக இருந்தாலும், Linux கெர்னலின் வரலாறு படிப்பதற்கு மிகவும் சுவாரஸ்யமாக இருக்கும். பின்வரும் கதையைப் படிப்பதன் மூலம் நீங்களே இதை உணர்வீர்கள்.

Linux கெர்னலின் வரலாறானது 20-ம் நூற்றாண்டின் 70-களிலிருந்து துவங்குகிறது. 1969-ம் ஆண்டு அமெரிக்காவிலுள்ள AT&T நிறுவனத்தின் subsidiary-ஆன Bell Laboratories தான் Unix எனும் இயக்குதளத்தை

முதன்முதலாக வெளியிட்டது. இந்த unix-தான் Linux உருவாவதற்கு அடிப்படைக் காரணமாக அமைந்தது.

Linux கெர்னலின் உருவாக்கத்தில் இரண்டு விஷயங்கள் பெரும்பங்கு வகிக்கின்றன. அவை GNU மற்றும் Minix ஆகும்.

GNU

1983-ம் ஆண்டு Richard M Stallman என்பவர் GNUஎனும்

திட்டத்தைத் துவக்கினார். இந்தத் திட்டத்தை துவக்கிய காலத்தில் அவர் Cambridge-ல் உள்ள Massachusetts Institute of Technologyஎனும் நிறுவனத்தில் Artificial Intelligence சோதனைக்கூடத்தில் பணிபுரிந்து கொண்டிருந்தார். அத்தகைய காலகட்டத்தில் Stallman தலைசிறந்து விளங்கிய நிரலாளர்களில் ஒருவர் ஆவார். மேலும் அவர் பணிபுரியும் இடத்தில் இருக்கும் நிரலாளர்கள் அனைவரும் அவரவர்களுடைய நிரல்களையும் அதன் source code-ஐயும் ஒருவருக்கொருவர் பரிமாறிக்கொண்டு கூட்டாக வேலைபார்த்து முன்னேறிக்கொண்டிருந்தனர். .

அப்போதுதான் AT&Tநிறுவனம் unix-க்கான உரிமத்தை $40,000 டாலர்களுக்கு விற்கத் தொடங்கியது. மேலும் “இந்த உரிமத்தை விலைகொடுத்து வாங்காத எந்த

நிரலாளருக்கும் இந்த unix-ன் source code-ஐ அவரவர் திட்டங்களுக்குப் பயன்படுத்த உரிமை இல்லை”

என்றும் அறிவித்தது. எனவே மிகப்பெரிய நிறுவனங்களால் மட்டுமே இதனை விலைகொடுத்து வாங்க முடிந்தது. இது Stallman-க்கு பேரதிர்ச்சியாக இருந்தது. ஒரு திட்டத்திற்குத் தேவையான source code ஏற்கனவே இருப்பினும் அதற்கான உரிமத்தை ஒரு நிறுவனம் பெற்றுக்கொண்டு அதனை மற்றவர்களுக்குத் தர மறுப்பதால் ஏற்கனவே இருக்கும் அந்த source code -ஐயே மீண்டும் உருவாக்கும் நிலைக்கு நிரலாளர்கள் தள்ளப்பட்டனர். இது ஓர் ஆரோக்கியமான வளர்ச்சியாக Stallman-க்குத் தோன்றவில்லை. ஏனெனில் ஒவ்வொருவரும் அவரவரது அறிவினைப் பரிமாறிக் கொள்வதற்குக் கூடப் பணம் கேட்கத் துவங்கினர்.

இந்த நிலையை மாற்றுவதற்காக 1983-ம் ஆண்டு Stallman தொடங்கிய திட்டமே GNU என்பதாகும். இந்தத் திட்டத்தின் நோக்கம் முழுக்க முழுக்க இலவசமான ஓர் இயக்குதளத்தை உருவாக்குவதே ஆகும்.

இதற்காக Stallman அனுப்பிய மின்னஞ்சல் பின்வருமாறு:

========

வியாழக்கிழமை,செப்டம்பர் 27, 1983 12:35:59 EST

நான் GNU (GNU is Not Unix) எனப்படும் Unix compatible software system-ஐ முழுவதுமாக எழுதி அதனை அனைவரும் பயன்படுத்தும் வகையில் இலவசமாக அளிக்கப் போகிறேன். இத்திட்டத்திற்கு விருப்பமுள்ள அனைவரும் பணம் வழங்கியோ, கருவிகள் வழங்கியோ,

அவரவர்களுடைய நிரல்களை வழங்கியோ அல்லது அவர்களுடைய நேரங்களை எங்களுக்காக செலவிட்டோ எங்களுக்கு உதவலாம்.

இந்த GNU திட்டத்தின் கீழ் kernel மற்றும் C Programs-ஐ எழுதி இயக்குவதற்குத் தேவையான அனைத்து utilities-ம் உருவாக்கப்படும். அவை editor, shell, C Compiler, linker, assembler மற்றும் இன்னும் சிலவற்றை உள்ளடக்கியது. இதன் பிறகு text editor, a YACC, an Empire game, spreadsheet போன்ற பல வகையான applications-ஐயும் உருவாக்க உள்ளோம். இது தவிர நாம் உருவாக்கப்போகும் ஒவ்வொன்றைப் பற்றியும் online மற்றும் hardcopy documentation-ஐயும் தரவிருக்கிறோம்.

GNU-ஆல் unix நிரல்களை இயக்க முடியும். ஆனால் இது

unix-க்கு ஒத்தவாறே இருக்காது. போகப்போக வேறுசில இயங்குதளங்களிலிருந்து நமக்கு கிடைக்கும் அனுபவத்தை வைத்து புதிது புதிதாக ஒவ்வொன்றையும் நமது இத்திட்டத்தின் கீழ் உருவாக்குவோம்.

==========

இவ்வாறாக அவர் மின்னஞ்சல் அனுப்பினார்.

GNU என்பது GNU is Not Unixஎனப் பொருள்படும். Unix எப்பொழுதுமே இலவசமாக வழங்கப்படாது. அதாவது அதன் மூல நிரல்களை நாம் பயன்படுத்த முடியாமல் நம்மைத் தடுப்பதன் மூலம் நமது சுதந்திரத்தையும் நமது கணிணியின் மீது நமக்கு இருக்கும் ஆளுமையையும் இழக்கச் செய்கிறது. இந்த விஷயத்தில் unix-ஐ தற்கால windows-வுடன்

ஒப்பிடலாம். சில காலங்களுக்குப் பின்னர், Stallman எழுதிய பிரபலமான Manifest GNU என்பதே தற்போது GPL (General Public License) என்றழைக்கப்படும் உரிமம் வருவதற்கு வித்தாக அமைந்தது.

GPL-ன் உயரிய குறிக்கோள் என்னவெனில் பயனர்கள் பின்வரும் நான்கு உரிமைகளைப் (சுதந்திரங்களைப்) பெற்றிருக்க வேண்டும் என்பதே ஆகும்.

- ஒரு நிரலை எத்தகைய தேவைக்கு வேண்டுமானாலும் பயன்படுத்தும் சுதந்திரம் (Freedom #0)

- ஒரு மென்பொருளின் மூல நிரலை அணுகி, அதனை நம்முடைய தேவைக்கு ஏற்றவாறு மாற்றும் சுதந்திரம் (Freedom #1)

- ஒன்றின் நிரலை மற்றவர்களுடன் பகிர்ந்து கொள்ளும் சுதந்திரம் (Freedom #2)

- ஒரு Program-ஐ மேம்படுத்தவும், அவ்வாறு மேம்படுத்தப்பட்ட programs-ஐ அதன் மூல நிரல்களுடன் சேர்த்து வெளியிடும் சுதந்திரம் (Freedom #3).

இதில் கடைசி சுதந்திரத்தை நாம் நன்றாகப் புரிந்து கொள்ள வேண்டும். நிரலாளர்கள் கட்டற்ற மென்பொருட்களை மேம்படுத்தி வர்த்தக ரீதியிலும் பயன்படுத்தலாம். ஆனால் அவ்வாறு மேம்படுத்தப்பட்ட மென்பொருட்களை மீண்டும் GPL உரிமத்தின் கீழ் தான் வெளியிட வேண்டும். அதாவது கட்டற்ற மென்பொருட்களை பயன்படுத்தி மென்பொருள் நிறுவனங்கள் உருவாக்கிய மென்பொருட்களை

விற்கும்போது, பயனர்களுக்கு அதன் மூல நிரல்களையும் சேர்த்தே கொடுக்க வேண்டும்.

1990-களில் GNU திட்டத்தின் கீழ் கட்டற்ற இயக்குதளத்தின் இயக்கத்திற்குத் தேவைப்படும் பெரும்பான்மையான components உருவாக்கப்பட்டு விட்டன. Emacs text editor, Compiler gcc, GNU C Compiler மற்றும் debugger gdb போன்றவற்றை stallman ஒருவரே உருவாக்கினார். இவர் ஒரு மிகச் சிறந்த நிரலாளர் என்பதால் அவர் உருவாக்கிய compiler மிகப் பெரிய நிறுவனங்களில் பல நிரலாளர்கள் சேர்ந்து உருவாக்கியவற்றை விட மிகவும் சிறப்பு வாய்ந்ததாகவும், தரமாகவும் அமைந்தது. பின்னர் இந்த compilers, பல்வேறு கணினி மொழிகளான C++, Pascal மற்றும் Fortran ஆகியவற்றுக்கும் உருவாக்கப்பட்டது. இப்போது GCC

என்பது GNU Compiler Collection எனப் பொருள்படும்.

Richard Stallman-ஐப் பொறுத்தவரை 1990-ல் GNU System-ஆனது முழுமைபெறும் தருவாயில் இருந்தது. ஆனால் ஓர் அடிப்படை component-ஆன kernel மட்டும் உருவாக்கப்படவில்லை. இந்த kernel-ஆனது (GNU Hurd என்று அழைக்கப்பட்டது) Mach-ல் இயங்கும் ஒரு set of server process-ஆக implement செய்யப்படும் என்று எதிர்பார்க்கப்பட்டது. Mach என்பது Carnegie Mellon பல்கலைக்கழகம் மற்றும் Utah பல்கலைக்கழகத்தால் உருவாக்கப்பட்ட ஒரு micro kernel ஆகும்.

ஆனால் இதன் உருவாக்கத்திற்கு மிகவும் காலதாமதம் ஆன நிலையில் kernel மட்டும் கிடைக்கப்பெறாமல் இருந்தது.

Minix:

1990-களில் கணினியானது Intel Microprocessor-ஐயும், Microsoft-ன் இயக்குதளத்தையும் அடிப்படையாகக் கொண்டே இருந்தது. Microsoft நிறுவனமானது desktop சந்தையிலும், server சந்தையிலும் ஆளுமை பெற்று விளங்கியது. Intel மற்றும் Intel compatible processors-ஐக் கொண்ட கணினியின் இயக்கும் திறனானது, unix-ஐக் கொண்டு இயங்கும் workstations-ன் திறனை எட்டியது. ஆனால் வர்த்தக ரீதியில் பயன்படுத்தப்படும் பெரும்பான்மையான unix based system-ன் versions, Intel-வன்பொருளில் இயங்கமுடியாதவை.

Unix-ஐ உருவாக்குபவர்கள் ஒரு சில குறிப்பிட்ட processors-ஐ (eg: SGI & MIPS) உருவாக்குபவர்களுடன் மட்டுமே இணைந்து செயல்பட்டனர். எனவே Unix-க்கு மாற்றாக Intel Processor-ல் இயங்க வல்ல ஒரு OS தேவைப்பட்டது. அதற்காக 1987-ம் ஆண்டு Andrew Tanenbaum என்பவர் Minix எனப்படும் ஒரு OS-ஐ உருவாக்கினார். இவர் Netherlands-ன் , Amsterdam நகரில்

உள்ள Vrije University-ல் கணினித்துறையின் பேராசிரியாராகப் பணிபுரிகிறார். இவர் உருவாக்கிய Minix-ஆனது மாணவர்களுக்கு ஒரு OS-ன் உள் அமைப்புகளைப்பற்றி விரிவாகக் கற்பிக்கப் பயன்பட்டது.

இந்த OS முழுமையாக வடிவமைக்கப்படாவிட்டாலும் இதனிடம் ஒரு சிறப்பு அம்சம் இருந்தது. C மற்றும் Assembly language-ல் எழுதப்பட்ட 12,000 வரிகள் கொண்ட இதன் மூல நிரல் வெளிப்படையாகக் கொடுக்கப்பட்டிருக்கும். அதாவது Tanenbaum-ஆல் எழுதப்பட்ட "Operating Systems: Design and Implementation" எனும் புத்தகத்தைப் படிக்கும் ஒவ்வொருவரும் Minix இயக்குதளத்தின் மூல நிரலை முழுமையாகப் படித்துப் புரிந்து கொள்ள முடியும். இவ்வாறாக Linus Torvalds இந்த Minix-ன் மூல நிரலில்

ஒருசில மேம்பாடுகளைச் செய்து Linux-ஐ உருவாக்கினார்.

Linux:

1991-ம் ஆண்டு Linus Benedict Torvalds எனும் Finlandமாணவர், Intel Processor-ஐக் கொண்ட அவரது கணினியில் Unix இயக்குதளத்தில் பொருந்துமாறு ஓர் kernel-ஐ எழுதுவதில் ஆர்வம் காட்டினார். இதற்கு Minixஎனும் இயக்குதளத்தை அவர் prototype-ஆக எடுத்துக் கொண்டார்.

அவர் வீட்டில் ஏற்கனவே பயன்படுத்திக் கொண்டிருந்த Minix-ல் ஒருசில மேம்பாடுகளைச் செய்து ஒரு புதிய kernel-ஐ உருவாக்கி அதனை GPL உரிமத்தின் கீழ் வெளியிட்டார். இவ்வாறாக Linus Torvalds உருவாக்கிய

புதிய kernel-க்கு அவர் பெயரே சூட்டப்பட்டு Linux என்றழைக்கப்பட்டது.

1991-ம் வருடம் August 25-ம் நாள், Minix news group-க்குப் பின்வருமாறு ஒரு செய்தி Torvalds-ஆல் அனுப்பப்பட்டது. அவர் அனுப்பிய செய்தி இங்கு தமிழில் மொழிமாற்றிக் கொடுக்கப்பட்டுள்ளது.

அவர் அனுப்பிய மின்னஞ்சல் பின்வருமாறு:

Minix பயன்படுத்திக் கொண்டிருக்கும் அனைவருக்கும் என்

அன்பார்ந்த வணக்கங்கள்!!!

நான் கடந்த மாதத்தில் இருந்து இலவச இயக்குதளம் உருவாக்கும் முயற்சியில் உள்ளேன். அதற்கான வேலைகளும் ஆரம்பிக்கப்பட்டு விட்டன. ஆனால் இது GNU போன்று ஒரு பெரிய திட்டம் கிடையாது. பொழுதுபோக்காக இதை நான் செய்யத் தொடங்கினேன். நான் உருவாக்கிக் கொண்டிருக்கும் இயக்குதளம் Minix-ன் வடிவமைப்பையே பெற்றிருக்கும். எனவே Minix-ல் நீங்கள் என்னென்ன மாற்றங்களைப் பெற விரும்புகிறீர்களோ அதை எனக்குத் தெரியப்படுத்தவும். அதனை நான் உருவாக்கிக் கொண்டிருக்கும் இந்தப் புதிய இயக்குதளத்தில் செயல்படுத்த முயல்கிறேன். ஆனால் உறுதியாக உங்களது விருப்பங்கள் இந்தப் புதிய இயக்குதளத்தில் இடம் பெறும் என்று கூற முடியாது.

தற்போது வரை bash(1.08) மற்றும் gcc(1.40)-ஐ port செய்து

விட்டேன். அவை நன்றாக செயல்பட்டுக் கொண்டிருக்கின்றன. உங்களது விருப்பங்கள் அனைத்தையும் எனக்கு torvalds@kruuna.helsinki.fi எனும் முகவரியில் தெரியப்படுத்தவும்.

================

இவ்வாறாக அவர் மின்னஞ்சல் அனுப்பினார்.

முதன்முதலில் Torvalds இந்தப் புதிய இயக்குதளத்தை Freax என்றே அழைத்தார் இந்தப் பெயர் kernel/makefile 0.11-லும் மற்றும் சில programs-ன் நிரலிலும் காணப்பட்டது. ஆனால் Ari lemmke என்பவர் FTP site-ல் இந்தப் புதிய system-ஐ load செய்வதற்கான directory-ஐ pub/OS/Linux என்றழைத்தார். இவ்வாறாக Linux என்ற பெயரே இந்தப்

புதிய OS-க்குப் பொருந்தி விட்டது.

1991-ல் Torvalds அவர் உருவாக்கிய கெர்னல்-க்கான மூல நிரலை வலைதளத்தில் பதிவு செய்தார். வலைத்தளம் மிகவும் பிரபலமடைந்து காணப்படாத அந்த காலகட்டத்திலேயே Torvalds-ன் புதிய இயக்குதளத்திற்கு மாபெரும் வரவேற்ப்பு கிடைத்தது.

பின்னர் 1992-ல் யாரெல்லாம் Linux கெர்னல்-ஐப் பயன்படுத்தி, பரிசோதனை செய்து பார்த்துள்ளீர்களோ அவர்கள் அதைப்பற்றிய கருத்துக்களை எனக்குக் கடிதம் மூலம் தெரியப்படுத்தவும் என்று செய்தி விடுத்தார்.

உலகத்தின் பெரும்பான்மையான பகுதிகளிலிருந்து Torvalds-க்கு கடிதங்கள் வந்து குவிந்தன. இதுவே Torvalds

மிகவும் பிரபலமடைந்துவிட்டார் என்பதைத் தெரிவித்தது. பின்னர் ஆயிரக்கணக்கில் தன்னார்வலர்கள் Linux-கெர்னல்க்கு வந்து குவிந்தனர். Linux கெர்னலும் நன்றாக வளர்ந்து கொண்டிருந்தது. Linux கெர்னல்-ஆனது GPL உரிமத்தின் கீழ் வெளியிடப்பட்டதால் அதன் மூல நிரலை தன்னார்வலர்கள் படித்து, அதன் மூல நிரலில் சற்று மாற்றம் செய்து Linux கெர்னலின் புதுப்புது versions-ஐ வெளியிட்டுக் கொண்டிருந்தனர்.

பின்னர் சீக்கிரமாகவே மென்பொருள் நிறுவனங்கள் Linux கெர்னலின் பக்கம் திரும்பினர்.

பல்வேறு மென்பொருட்கள் ஒன்றாக compile செய்யப்பட்டு distribution format-ல் பயனர்களுக்கு விற்பதன் மூலம் வணிகர்கள் அதனை வர்த்தக ரீதியில் பயன்படுத்தினர்.

Linux கெர்னலும் GNU மென்பொருட்களும் GNU/Linux Distribution என்று அழைக்கப்படுகிறது.

Red Hat, Caldera போன்ற நிறுவனங்கள் இதை வைத்து உலகமெங்கும் வியாபாரம் செய்து பயனர்களைப் பெருக்கினர். Debian என்பது தன்னார்வத் தொண்டர்களால் உருவாக்கப்பட்ட ஒரு distribution ஆகும். நாளடைவில் X-windows system, KDE, GNOME போன்றவை GUI-ஐப் பெற்றிருக்கும் distributions-ஆக மிகவும் பிரபலமடைந்து வந்தது.

ஆரம்ப காலகட்டத்தில் Torvalds அவரது படைப்புகளை விற்பதற்க்கு ஆர்வம் காட்டவில்லை. இது அவரது முதல் version 0.01-ன் acknowledgement மற்றும் copyrights-ல் தெளிவாகக் குறிப்பிடப்பட்டுள்ளது. சொல்லப்போனால்,

அவர் GNU license-ன் விதிமுறைகளைக் காட்டிலும் சற்று கடினமான விதிமுறைகளை விதித்தார். அதாவது Linux-ஐப் பயன்படுத்துவதற்கோ மற்றவர்களுடன் பகிருவதற்கோ பணம் வாங்கக் கூடாது எனும் கடும் விதிமுறையை விதித்தார்.

ஆனால் 1992-ம் ஆண்டு February மாதம் சிலர் GNU/Linux distributions-ஐ floppy disk-ல் பதிவுசெய்து மற்றவர்களுக்குக் கொடுக்கும்போது அதற்காக நாம் செலவிடும் நேரத்துக்கும், floppy disk-க்கு ஆகும் செலவுக்கு மட்டுமாவது பணம் பெற்றுக்கொள்ளலாமே என்று அவர்களது விருப்பங்களைத் தெரிவித்தனர்.

மேலும் linux கெர்னல் மட்டும் பயனர்களுக்குத் தனியாக வழங்கப்படப்போவதில்லை. வலைதளத்தில் GNU திட்டத்தின் கீழ் ஆயிரக்கணக்கில் கொட்டிக்கிடக்கும்

மென்பொருட்களுடன் சேர்த்தே GNU/Linux-ஆக பயனர்களுக்கு வழங்கப்படுகிறது. GCC Compiler-தான் linux கெர்னலின் செயல்பாட்டிற்கு மிகவும் முக்கியமானது. ஆனால் இது GPL உரிமத்தின் கீழ் வரக்கூடிய ஒன்றாகும். எனவே Torvalds தனது அடுத்த version 0.12-ஐ GPL உரிமத்தின் கீழ் வெளியிட்டார்.

இவ்வாறாக GNU/Linux ஆனது படிப்படியாக வளர்ச்சியடைந்து இன்று உலகம் முழுவதும் கோடிக்கணக்கான தன்னார்வலர்கள் GNU/Linux-ன் வளர்ச்சிக்காக உழைத்துக் கொண்டே இருக்கின்றனர். GNU/Linux என்பது மனித சக்தியின் ஒரு மாபெரும் கூட்டு முயற்சியே ஆகும்.

Users-ஐ கையாளுதல்

இந்தப் பகுதியில் நாம் user management-க்கு உதவும் ஒருசில commands-ஐப் பற்றி விரிவாகக் காண்போம்.

root என்றால் என்ன?

Ubuntu-வை install செய்யும்போது அனைத்து வகையான சிறப்பு அனுமதிகளையும் பெற்றிருக்கும் root account-ஆனது உருவாக்கப்படும். மேலும் இந்த root-ஆனது தானாகவே disable செய்யப்பட்ட நிலையில்

இருக்குமாறு ubuntu developers அதனை உருவாக்கி உள்ளனர். எனவே சாதாரண பயனர்களால் root-ஐப் பயன்படுத்த முடியாது. வேண்டுமானால் sudo எனும் tool-ன் உதவியுடன் இதனை நாம் பயன்படுத்தலாம். sudo-ஆனது ஒரு சாதாரண பயனருக்கு root-க்கான அனுமதிகளை வழங்கப் பயன்படுகிறது.

Ubuntu-வை install செய்பவரால் உருவாக்கப்படும் முதல் பயனர் /etc/sudoers எனும் file-ல் தானாகவே இணைக்கப்பட்டு விடுவார். வேறு ஏதாவது ஒரு பயனருக்கு sudo அனுமதிகளை வழங்க விரும்பினால் அவரையும் இந்த admin group-ல் இணைத்து விட்டால் போதுமானது.

sudo command

sudo-ஆனது நம்மை root user-ஐப் போன்று செயல்பட வைக்கும் ஒரு command ஆகும்.

இதற்கு முதலில் root-ஆனது நமக்கு sudo-வை பயன்படுத்துவதற்கான அனுமதிகளை வழங்கியிருக்க வேண்டும். அதன் பின்னர் தான் நாம் sudo-வை வைத்து, root user செய்யும் வேலைகளை செய்ய முடியும்.

உதாரணத்துக்கு root-க்குத் தான் நமது கணினியில் ஒரு புதிய பயனரை உருவாக்கும் அதிகாரம் உள்ளது. ஆனால் நாம் இந்த sudo-ன் துணை கொண்டு ஒரு புதிய பயனரை உருவாக்க முடியும். இதனை நாம் பின்வரும் adduser command-ல்

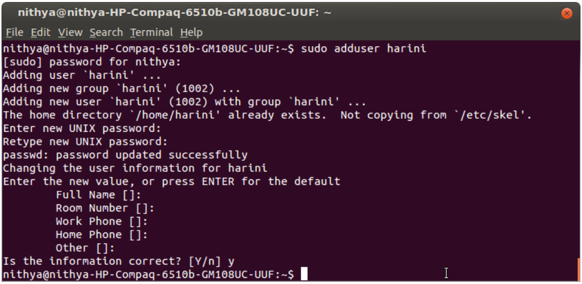
பார்க்கலாம்.

adduser command

sudo மூலம் இந்த adduser command-ஐ நாம் இயக்கும்போது நமது கணினியில் ஒரு புதிய user-ஐ நாம் உருவாக்க முடியும். இது பின்வருமாறு.

```
$ adduser harini
```

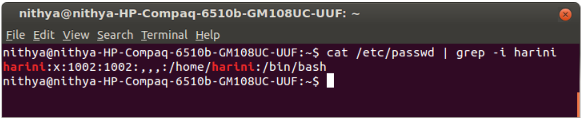
இதன் தொடர்ச்சியாக கேட்கப்படும் கேள்விகளுக்கு நாம் சரியான விவரங்களை அளித்த பின்னர், harini எனும் பெயரில் ஒரு புதிய user உருவாக்கப்படும்.



```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~
File Edit View Search Terminal Help
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~$ sudo adduser harini
[sudo] password for nithya:
Adding user `harini' ...
Adding new group `harini' (1002) ...
Adding new user `harini' (1002) with group `harini' ...
The home directory `/home/harini' already exists.  Not copying from `/etc/skel'.
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for harini
Enter the new value, or press ENTER for the default
        Full Name []:
        Room Number []:
        Work Phone []:
        Home Phone []:
        Other []:
Is the information correct? [Y/n] y
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~$
```

பொதுவாக ஒரு கணினியில் உள்ள users பற்றிய விவரங்கள் அனைத்தும் /etc/passwd எனும் பகுதியில் சேமிக்கப்படும். எனவே இந்தப் பகுதியில் நாம் உருவாக்கிய harini எனும் user இணைக்கப்பட்டுவிட்டதா என்பதைப் பின்வருமாறு சரிபார்க்கலாம்.

$ cat /etc/passwd



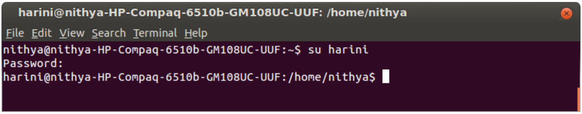
இங்கு password எனுமிடத்தில் வெறும் x உள்ளதைக் காணலாம். /etc/shadow எனும் இடத்தில் மட்டுமே அனைத்து users-ன் encrypt செய்யப்பட்ட password-ம் காணப்படும். இது பின்வருமாறு.

$ sudo cat /etc/shadow

su command

su ஆனது switch user எனப் பொருள்படும். su மூலம் ஒரு பயனர் நமது கணிணியில் ஒரு பயனர் கணக்கில் இருந்து வேறு ஒரு பயனரின் கணக்கிற்கு எளிதில் மாற முடியும். இது நாம் logout செய்த பின்னர் மற்றொருவர் login செய்வதைக் காட்டிலும் சுலபமானது.

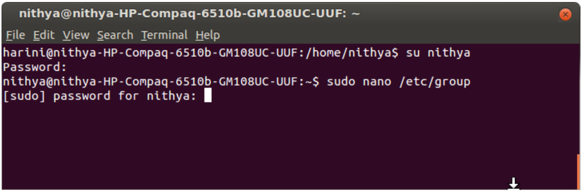
$ su harini



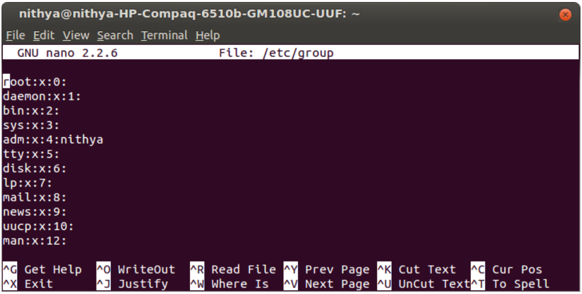
ஒரு சாதாரண user-க்கு sudo அனுமதிகளை வழங்குதல்

root user அல்லது sudo அனுமதிகளைப் பெற்ற ஒரு user,

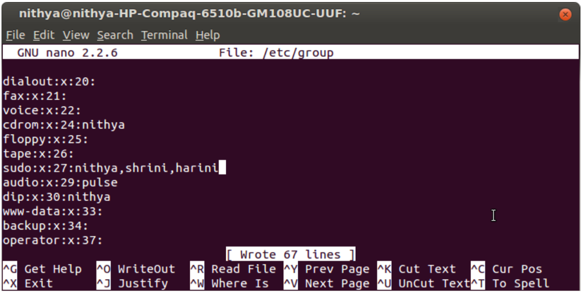
ஒரு சாதாரண user–க்கு sudo அனுமதிகளை வழங்க முடியும். உதாரணத்துக்கு harini-க்கு நாம் sudo அனுமதிகளை விரும்பினால், /etc/group-க்குள் harini-யை நாம் இணைக்க வேண்டும். இது பின்வருமாறு.



இங்கு nano என்பது vi-ஐப் போன்று செயல்படும் ஒரு editor ஆகும். எனவே nano-வை கொண்டு ஒரு file-ஐ open செய்யும்போது பின்வருமாறு ஒரு திரை வெளிப்படும்.



இதில் arrow mark key-ஐப் பயன்படுத்தி sudo-க்கான entry வரும்வரை நகர்ந்துகொண்டே செல்லவும். sudo-க்கான entry வந்தவுடன் அந்த வரியின் இறுதியில் சென்று comma-வை இணைத்து ‘harini’ என type செய்துவிட்டு ctrl+O வை அழுத்தவும். இது நாம் செய்த மாற்றங்களை file-ல் overwrite செய்யப் பயன்படும்.



இப்போது harini-க்கு sudo அனுமதிகள்

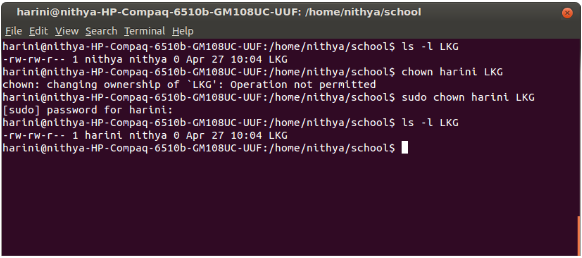
வழங்கப்பட்டுவிட்டது

chown command

chown command-ஆனது ஒரு file-ன் owner-ஐ மாற்றி

அமைக்க உதவும். உதாரணத்துக்கு LKG file-ன் user, ‘Nithya’ ஆவார். இதனை ‘Harini’ என்று மாற்றி அமைக்க விரும்பினால், command-ஐ பின்வருமாறு அமைக்கவும்.

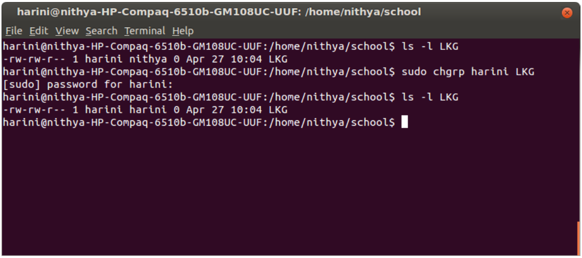
$ chown harini LKG



chgrp command

chgrp command-ஆனது ஒரு file-ன் group-ஐ மாற்றி அமைக்க உதவும். உதாரணத்துக்கு LKG file-ன் group, ‘Nithya’ ஆகும். இதனை ‘Harini’ என்று மாற்றி அமைக்க விரும்பினால், command-ஐ பின்வருமாறு அமைக்கவும்.

```
$ chgrp harini LKG
```



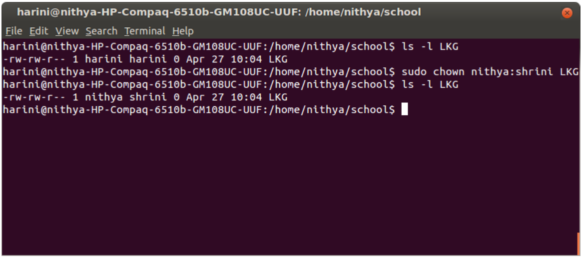
ஒரே நேரத்தில் owner மற்றும் group-ஐ மாற்றுதல்

chmod command மூலம் நாம் ஒரே நேரத்தில் owner மற்றும்

group-ஐ எவ்வாறு மாற்றுவது என்று பின்வருமாறு

பார்க்கலாம்.

$ chmod nithya:shrini LKG



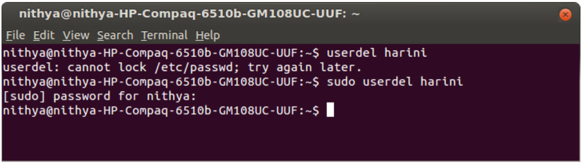
இது owner-ஐ nithya-ஆகவும், group-ஐ shrini-ஆகவும்

மாற்றுகிறது. இங்கு நாம் இடையில் colon-ஐப் பயன்படுத்துவதற்கு பதிலாக dot-ஐக் கூடப் பயன்படுத்தலாம்.

userdel Command

userdel command-ஆனது ஒரு user-ஐ கணிணியில் இருந்து நீக்கப் பயன்படுகிறது.

```
$ userdel harini
```



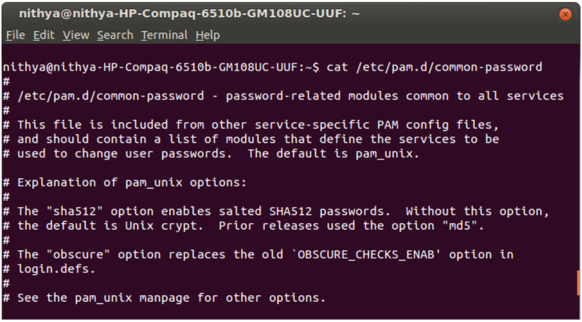
Password Policy:

ஒரு password எவ்வாறு அமைய வேண்டும் என்பதற்கு ஒரு சில விதிமுறைகள் உள்ளன. அவற்றைப்பற்றி இந்தப் பகுதியில் காணலாம். ஒரு password ஆனது குறைந்தபட்சம் 4 எழுத்துக்களைக் கொண்டதாகவும், ஒருசில அடிப்படை entropy checks-ஐப் பெற்றும் அமையும்.

அதாவது இந்த விவரங்கள் அனைத்தும் /etc/pam.d

/common-password எனும் file-ல் பின்வருமாறு

காணப்படும்.



```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~
File Edit View Search Terminal Help
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~$ cat /etc/pam.d/common-password
#
# /etc/pam.d/common-password - password-related modules common to all services
#
# This file is included from other service-specific PAM config files,
# and should contain a list of modules that define the services to be
# used to change user passwords.  The default is pam_unix.

# Explanation of pam_unix options:
#
# The "sha512" option enables salted SHA512 passwords.  Without this option,
# the default is Unix crypt.  Prior releases used the option "md5".
#
# The "obscure" option replaces the old `OBSCURE_CHECKS_ENAB' option in
# login.defs.
#
# See the pam_unix manpage for other options.
```

password செயலிழக்கும் தருணம்:

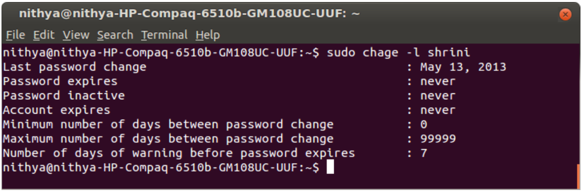
ஒரு user-ஐ நாம் உருவாக்கும் பொழுதே, அவரின்

password-க்கான வாழ்நாளின் குறைந்தபட்ச மற்றும்

அதிகபட்ச கால அளவு நிர்ணயிக்கப்படுகிறது. இந்தக் கால அளவிற்குள் அவர் தனது password-ஐ மாற்றி அமைக்க வேண்டும்.

ஒரு பயனருடைய password-ன் வாழ்நாளைப் பற்றித் தெரிந்து கொள்ள பின்வரும் command பயன்படுகிறது.

```
$ sudo chage -l shrini
```



மேலும் இவற்றை மாற்ற command-ஐ பின்வருமாறு அமைக்கவும்.

$ sudo chage shrini

பின்னர் இதன் தொடர்ச்சியாக கேட்கப்படும் கேள்விகளுக்குப் பதிலளிப்பதன் மூலம் நமக்குத் தேவையான வகையில்

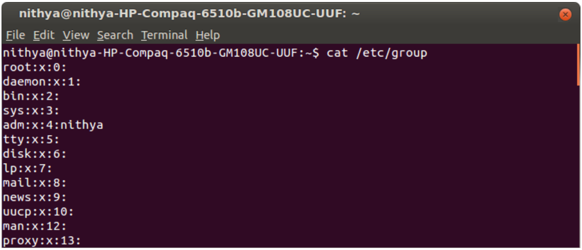
password-ன் வாழ்நாளை மாற்றி விடலாம். பின்னர் sudo chage -l shrini எனக் கொடுப்பதன் மூலம் நாம் மாற்றியதை சரிபார்த்துக் கொள்ளலாம்.

```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~
File Edit View Search Terminal Help
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~$ sudo chage shrini
Changing the aging information for shrini
Enter the new value, or press ENTER for the default

        Minimum Password Age [0]: 20
        Maximum Password Age [99999]: 30
        Last Password Change (YYYY-MM-DD) [2013-05-13]: 2013-07-10
        Password Expiration Warning [7]: 8
        Password Inactive [-1]:
        Account Expiration Date (YYYY-MM-DD) [1969-12-31]: 2015-01-01
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~$ sudo chage -l shrini
Last password change                                    : Jul 10, 2013
Password expires                                        : Aug 09, 2013
Password inactive                                       : never
Account expires                                         : Jan 01, 2015
Minimum number of days between password change          : 20
Maximum number of days between password change          : 30
Number of days of warning before password expires       : 8
```

Groups

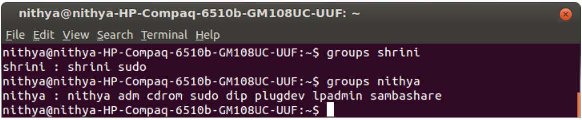
நமது கணிணியில் உள்ள அனைத்து groups-ஐயும் பட்டியலிட பின்வரும் command பயன்படுகிறது. இது நமது கணிணி பயன்படுத்தும் groups உட்பட அனைத்து groups-ஐயும் பட்டியலிடும்.

$ cat /etc/group



ஒரு பயனர் எந்தெந்த groups-ல் உள்ளார் என்பதைக் கண்டறிய பின்வரும் command பயன்படுகிறது.

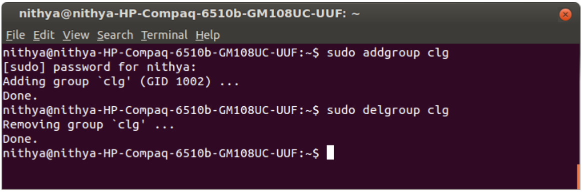
$ groups shrini



ஒரு group-ஐ உருவாக்குவதற்கும் அழிப்பதற்கும் பின்வரும் commands பயன்படுகிறது.

$ sudo addgroup clg

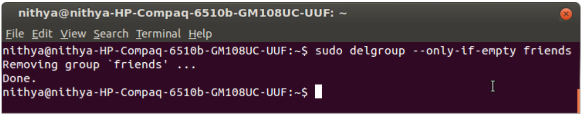
$ sudo delgroup clg



இங்கு clg எனும் group உருவாக்கப்பட்டிருப்பதையும், அழிக்கப்பட்டிருப்பதையும் காணலாம்.

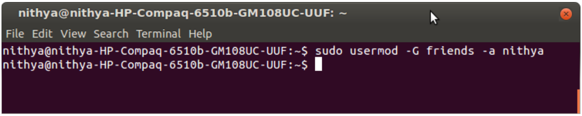
காலியாக இருந்தால் மட்டுமே group-ஐ நீக்க:

$ sudo delgroup –only-if-empty *group*



ஒரு user-ஐ ஒரு group-ல் சேர்த்தல்:

$ sudo usermod -G *group* -a *user*



User ஐ நீக்குதல்:

நீக்கும் முன் user-ன் தகவல்களை backup செய்தல்.

$ sudo deluser –backup-to /location/to/put/backup/user

–remove-home *user*

User-ஐ நீக்குதல்

user-ஐயும் home directory ஐயும் சேர்த்து அழித்தல்

```
$ sudo deluser –remove-home user
```

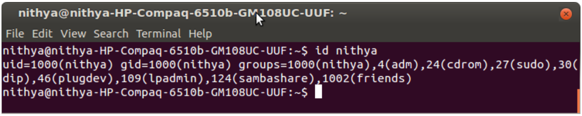
user, home directory மற்றும் user-க்கு ownership உள்ள எல்லா file-களையும் அழித்தல்

$ sudo deluser –remove-all-files *user*

Group-ஐ கண்டறிதல்

இது ஒரு user-ன் Group-களைப் பற்றி அறிய உதவுகிறது.

$ id *user*

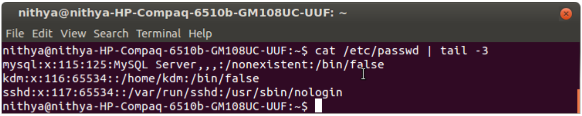


/etc/passwd file

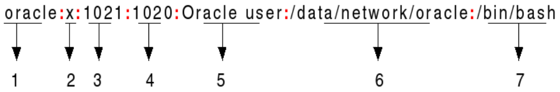
எல்லா பயனர்களின் தகவல்களும் /etc/passwd என்ற text fileல் சேமிக்கப்படுகின்றன. இந்த file-ஐ அனைவரும் படிக்கலாம். ஆனால் root மட்டுமே எழுத முடியும். இதில்

system user மற்றும் human user என எல்லா பயனரின் தகவல்களும் இருக்கும்.

/etc/passwd-ன் அமைப்பு:



இதில் ஒவ்வொரு பயனருக்கும் ஒரு வரி தரப்படுகிறது. ஒவ்வொரு file-ம் : மூலம் பிரிக்கப்படுகிறது. மொத்தம் 7 பகுதிகள் உள்ளன. இதன் அமைப்பு



1. **username** : பயனர் பெயர்
2. **password** : கடவுச்சொல். இது **x** மூலம் குறிப்பிடப்படுகிறது. **/etc/shadow** என்ற **file**-ல் கடவுச்சொல் சேமிக்கப்படுகிறது.

3. **User id** (**UID**) : ஒவ்வொரு பயனருக்கும் ஒரு **user id** (பயனர் எண்) தரப்படும். 0 என்பது **root**-க்கு ஒதுக்கப்பட்டுள்ளது. 1-999 வரை **system user**-க்கும்

அதற்குமேல் human user-க்கும் தரப்படுகிறது.

4. Group id (GID) : குழு எண், முதன்மைக் குழு எண் (Primary group id /etc/group) -ல் இருக்கும்.

5. User Id info: பயனர் விவரங்கள். இதில் பயனரின் சுய விவரங்களை தரலாம். முகவரி, தொலைபேசி எண், மின்னஞ்சல் போன்றவற்றை தரலாம். finger எனும் மென்பொருள் மூலம் இந்த விவரங்களை பெறலாம்.

6. home directory: பயனரின் home directory (இல்லம்)75

இதில் தான் பயனர் தமது கோப்புகளை சேமிக்க முடியும்.

7. **command / shell**: **login** செய்தவுடன் இயக்கப்பட வேண்டிய **command** அல்லது **shell** பொதுவாக **/bin/bash**. இதற்கு பதில் வேறு **command** கூட தரலாம்.

எல்லா பயனரையும் பார்க்க

```
$ less /etc/passwd
```

ஒரு பயனரை மட்டும் பார்க்க

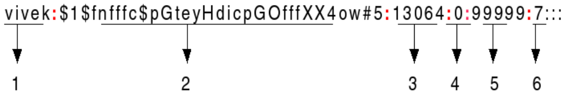
```
$ grep nithya /etc/passwd
```

/etc/shadow file:

/etc/passwd-ல் நமது கடவுச் சொல் சேமிக்கப்படுவதில்லை.

மாறாக /etc/shadow எனும் file-ல் சேமிக்கப்படுகிறது.

இதில் கடவுச் சொல் **encryption** செய்யப்பட்டு சேமிக்கப்படுகிறது. மேலும் **root** பயனரால் மட்டுமே இதைப் படிக்க முடியும். இதிலும் கூறுகள் : மூலமே பிரிக்கப்படுகின்றன.



1. **Username** : பயனர் பெயர்
2. **Password** : கடவுச் சொல் (**encryption** வடிவில்)

3. **last password change** : **Jan** 1, 1970ல் இருந்து கடைசியாக கடவுச் சொல் மாற்றப்பட்ட நாட்கள்

4. **Minimum** : கடவுச் சொல் மாற்றுவதற்கான குறைந்த பட்ச நாட்கள். இதற்குள் மாற்றிவிட்டால் நல்லது.

5. **Maximum** : கடவுச் சொல்லை மாற்ற அதிகபட்ச நாட்கள். இதற்குப் பின் கண்டிப்பாக மாற்றியே ஆக வேண்டும்.

6. **Warn** : கடவுச்சொல் மாற்ற எச்சரிக்கை தரும் நாட்கள்

7. **Inactive** : கடவுச்சொல்லை மாற்றாவிட்டால், கணக்கை முடக்கும் நாட்கள்

8. **expire** : கணக்கு முழுதும் முடங்கும் நாள்

இதில் கடவுச் சொல் **md**5 என்ற **algorithm** முறையில் **encryption** செய்யப்பட்டு சேமிக்கப்படுகிறது.

Disk Management

Hard disk ஆனது லினக்சில் பின்வருமாறு கருதப்படுகிறது.

/dev/sda : Primary Master

/dev/sdb : Primary slave

/dev/sdc : Secondary Master

/dev/sdd : Secondary slave

இயல்பாக உள்ள hard disk-ஆனது /dev/sda ஆகும்.

இதில் பல partition களை உருவாக்கலாம்.

/dev/sda1

/dev/sda2

/dev/sda3

/dev/sda4

...

பொதுவாக 4 primary partition மட்டுமே உருவாக்க முடியும். இதில் 4வதை extended ஆக மாற்றி, அதில் பல logical partition-களை உருவாக்கலாம்.

fdisk எனும் மென்பொருள் மூலம், இந்த partition-களை உருவாக்குதல் மற்றும் நீக்கல் போன்ற செயல்களை செய்யலாம்.

எச்சரிக்கை : கீழ் வருபவை உதாரணங்கள் மட்டுமே. root பயனரால் மட்டுமே இயக்கப்பட வேண்டும். நன்கு புரிந்து கொண்ட பின்பே, தேவையான போது மட்டுமே இயக்கவும். தவறாகிப் போனால், தகவல் இழப்பு ஏற்படும். தகவலை திருப்பி மீட்க முடியாது.

Partition-களை பார்த்தல்:

ஏற்கெனவே உள்ள partition-களை பார்வையிட

$ sudo fdisk -l

```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~
File  Edit  View  Search  Terminal  Help
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~$ sudo fdisk -l

Disk /dev/sda: 250.1 GB, 250059350016 bytes
255 heads, 63 sectors/track, 30401 cylinders, total 488397168 sectors
Units = sectors of 1 * 512 = 512 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0xb6355c82

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1   *        2048      206847      102400    7  HPFS/NTFS/exFAT
/dev/sda2          206848   283596799   141694976    7  HPFS/NTFS/exFAT
/dev/sda3       283598846   428128255    72264705    5  Extended
/dev/sda4       283596800   283598845        1023   83  Linux
/dev/sda5       283598848   291409919     3905536   82  Linux swap / Solaris
/dev/sda6       291411968   350003199    29295616   83  Linux
/dev/sda7       350005248   428128255    39061504   83  Linux

Partition table entries are not in disk order
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~$
```

1. ஏதேனும் ஒரு disk-ல் உள்ள partition-களை மட்டும் பார்வையிட

```
$ sudo fdisk -l /dev/sda
```

fdisk-ன் உள்ளே:

இது பல்வேறு, partition தொடர்பான கட்டளைகளை இயக்க உதவுகிறது. அனைத்து கட்டளைகளையும் பார்க்க. m

```
# fdisk /dev/sda
```

```
The number of cylinders for this disk is set to 9729.
There is nothing wrong with that, but this is larger t
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions o
2) booting and partitioning software from other OSs
  (e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): m
Command action
  a  toggle a bootable flag
  b  edit bsd disklabel
  c  toggle the dos compatibility flag
  d  delete a partition
  l  list known partition types
  m  print this menu
  n  add a new partition
```

o create a new empty DOS partition table

p print the partition table

q quit without saving changes

s create a new empty Sun disklabel

t change a partition's system id

u change display/entry units

v verify the partition table

w write table to disk and exit

x extra functionality (experts only)

1. ஒரு partition-ஐ நீக்குதல்

fdisk /dev/sda

The number of cylinders for this disk is set to 9729.

```
There is nothing wrong with that, but this is larger t
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions o
2) booting and partitioning software from other OSs
  (e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): p

Disk /dev/sda: 80.0 GB, 80026361856 bytes
255 heads, 63 sectors/track, 9729 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Disk identifier: 0xf6edf6ed

 Device Boot    Start       End    Blocks  Id System
/dev/sda1           1      1959  15735636   c W95 FAT32 (LBA)
/dev/sda2        1960      5283  26700030   f W95 Ext'd (LBA)
```

/dev/sda3	5284	6528	10000462+	7	HPFS/NTFS
/dev/sda4	6529	9729	25712032+	c	W95 FAT32 (LBA)
/dev/sda5 *	1960	2661	5638752	83	Linux
/dev/sda6	2662	2904	1951866	83	Linux
/dev/sda7	2905	3147	1951866	83	Linux
/dev/sda8	3148	3264	939771	82	Linux swap / Sol
/dev/sda9	3265	5283	16217586	b	W95 FAT32

Command (m for help): d

Partition number (1-9): 8

Command (m for help): d

Partition number (1-8): 7

Command (m for help): d

Partition number (1-7): 6

```
Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with e
The kernel still uses the old table. The new table wil
the next reboot or after you run partprobe(8) or kpart
Syncing disks.
```

3. புது partition-ஐ உருவாக்குதல்

இதன் மூலம் புது partition-களை தேவையான,

தொடக்க cylinder எண், partition-ன் அளவு (அ)

கடைசி cylinder எண் தந்து உருவாக்கலாம்.

உருவாக்கிய பிறகு p, தந்து print செய்யலாம்.

```
# fdisk /dev/sda

The number of cylinders for this disk is set to 9729.
There is nothing wrong with that, but this is larger t
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions o
2) booting and partitioning software from other OSs
  (e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): n
First cylinder (2662-5283, default 2662):
```

```
Using default value 2662
Last cylinder, +cylinders or +size{K,M,G} (2662-3264, def
Using default value 3264
```

கடைசியாக w தந்து மாற்றங்களை சேமிக்க வேண்டும்.

```
Command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with e
The kernel still uses the old table. The new table wil
the next reboot or after you run partprobe(8) or kpart
Syncing disks.
```

பிறகு mkfs மூலம் format செய்யலாம்.

mkfs.ext3 /dev/sda7

4. boot flag-ஐ மாற்றுதல்

ஒரு partition-ன் boot தன்மையை a இதன் மூலம் மாற்றலாம். தெரியாமல் இதை மாற்றி விட வேண்டாம்.

```
# fdisk /dev/sda

The number of cylinders for this disk is set to 9729.
There is nothing wrong with that, but this is larger t
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions o
2) booting and partitioning software from other OSs
  (e.g., DOS FDISK, OS/2 FDISK)
```

```
Command (m for help): p

Disk /dev/sda: 80.0 GB, 80026361856 bytes
255 heads, 63 sectors/track, 9729 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Disk identifier: 0xf6edf6ed

   Device Boot      Start         End      Blocks   Id  System
/dev/sda1               1        1959    15735636    c  W95 FAT32 (LBA)
/dev/sda2            1960        5283    26700030    f  W95 Ext'd (LBA)
/dev/sda3            5284        6528    10000462+   7  HPFS/NTFS
/dev/sda4            6529        9729    25712032+   c  W95 FAT32 (LBA)
/dev/sda5   *        1960        2661     5638752   83  Linux
/dev/sda6            3265        5283    16217586    b  W95 FAT32
/dev/sda7            2662        3264     4843566   83  Linux
```

```
Partition table entries are not in disk order

Command (m for help): a
Partition number (1-7): 5

Command (m for help): p

Disk /dev/sda: 80.0 GB, 80026361856 bytes
255 heads, 63 sectors/track, 9729 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Disk identifier: 0xf6edf6ed

   Device Boot    Start       End    Blocks  Id System
/dev/sda1              1      1959  15735636   c W95 FAT32 (LBA)
/dev/sda2           1960      5283  26700030   f W95 Ext'd (LBA)
```

```
/dev/sda3    5284    6528  10000462+  7  HPFS/NTFS
/dev/sda4    6529    9729  25712032+  c  W95 FAT32 (LBA)
/dev/sda5    1960    2661   5638752  83  Linux
/dev/sda6    3265    5283  16217586   b  W95 FAT32
/dev/sda7    2662    3264   4843566  83  Linux

Partition table entries are not in disk order

Command (m for help):
```

4. partition order-ஐ சரி செய்தல்

ஒரு சில logical partition-களை நீக்கி விட்டு, மீண்டும்

புதிதாய் உருவாக்கும் போது, பின்வரும் பிழை ஏற்படலாம். Partition out of order. Partition table entries are not in disk order.

Sda6, sda7, sda8 ஐ நீக்குவதாய் கொள்வோம். இப்போது புதிதாய் உருவாக்கும்போது sda6 என பெயரிடாமல், sda7 என பெயரிடப்படும். sda9 ஏற்கனவே இருந்தால், அது sda6 ஆகி விடும். இந்த பிழையை சரி செய்ய f பயன்படுகிறது. இனி இந்த பிழைச் செய்தி வராது.

```
$ fdisk /dev/sda

The number of cylinders for this disk is set to 9729.
There is nothing wrong with that, but this is larger t
and could in certain setups cause problems with:
1) software that runs at boot time (e.g., old versions o
2) booting and partitioning software from other OSs
  (e.g., DOS FDISK, OS/2 FDISK)

Command (m for help): p

Disk /dev/sda: 80.0 GB, 80026361856 bytes
255 heads, 63 sectors/track, 9729 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Disk identifier: 0xf6edf6ed
```

```
   Device Boot    Start       End    Blocks   Id  System
/dev/sda1             1      1959  15735636    c  W95 FAT32 (LBA)
/dev/sda2          1960      5283  26700030    f  W95 Ext'd (LBA)
/dev/sda3          5284      6528  10000462+   7  HPFS/NTFS
/dev/sda4          6529      9729  25712032+   c  W95 FAT32 (LBA)
/dev/sda5   *      1960      2661   5638752   83  Linux
/dev/sda6          3265      5283  16217586    b  W95 FAT32
/dev/sda7          2662      3264   4843566   83  Linux
```

Partition table entries are not in disk order

Command (m for help): x

Expert command (m for help): f

Done.

```
Expert command (m for help): w
The partition table has been altered!

Calling ioctl() to re-read partition table.

WARNING: Re-reading the partition table failed with e
The kernel still uses the old table. The new table wil
the next reboot or after you run partprobe(8) or kpart
Syncing disks.
```

Mount மற்றும் Unmount

Partition-ஐ உருவாக்குதல் என்பது ஒரு வீட்டில் ஒரு அறையை உருவாக்குதல் போல.

அதை பயன்படுத்த ஏதேனும் ஒரு file system ஐ உருவாக்க வேண்டும். ext3, ext4, ReiserFS, Btrfs போன்றவை குனு/லினக்சில் பொதுவாக பயன்படும் filesystems.

இவை Journalins Filesystem என்றும் அழைக்கப்படுகின்றன. ஒரு பத்திரிகையில் உள்ள கட்டுரைகளுக்கு index பக்கங்கள் உள்ளது போல், இவற்றில் சேமிக்கப்படும் file களுக்கு ஒரு index உருவாக்கப்படுகிறது. இதனால் வேகமும், நிலைப்புத் தன்மையும், hard disk-ன் ஆயுளும் அதிகரிக்கின்றன.

mkfs எனும் கட்டளை மூலம் file system உருவக்கலாம்.

```
sudo mkfs.ext4 /dev/sda5

sudo mkfs.reiserf5 /dev/sda6
```

இவ்வாறு உருவாக்கிய file systemகளை பயன்படுத்த அவற்றை mount செய்ய வேண்டும்.

mount என்பது raw-ஆக உள்ள device/partition ஐ அனைவரும் பயன்படுத்தும் வகையில் / க்கு அடியில் ஒரு folder உடன் இணைக்கும் செயல் ஆகும்.

Mount மூலம் disk partition, Linux network share, windows network share, iso file மற்றும் cd/dvd போன்றவற்றை சாதாரண folder போலவே பயன்படுத்த முடியும்.

Mount-ஐ பயன்படுத்துதல்:

இதை root பயனர் மட்டுமே செய்ய முடியும் அல்லது sudo அனுமதி பெற்ற பயனரும் செய்யலாம்.

உதாரணம் 1:

```
sudo mount /dev/sda5 /songs
```

இதில் /dev/sda5 ஆனது /songs-ல் mount செய்யப் படுகிறது. ls /songs அல்லது cd /songs மூலமாக /dev/sda5 ல் உள்ள file-களை அணுகலாம்.

உதாரணம் 2:

sudo mount -t iso9660 movie.iso /media/movie

இதில் movie.iso என்ற ISO file-ஐ /media/movie என்ற folder-ல் mount செய்கிறோம்.

எந்த ஒரு folder-லிலும் mount செய்யலாம். இது mount point எனப்படும்.

இதுவரை mount ஆகி உள்ள அனைத்து filesystem-களையும் காண, வெறும் mount என்ற கட்டளை உதவுகிறது.

Unmount செய்தல்:

ஏற்கெனவே mount செய்யப்பட்ட ஒரு filesystem ஐ விடுவிக்கும் செயல் unmount எனப்படும்.

இதற்கும் root- ன் அனுமதி தேவை. ஏற்கெனவே mount செய்யப்பட்ட file அல்லது folder வேறு எங்காவது பயனில் இருந்தாலோ, திறந்து இருந்தாலோ unmount செய்ய முடியும்.

```
sudo umount /media/movie
```

/etc/fstab

ஒரு கணிணி **boot** ஆகும்போதே பல **filesystem**-கள் தமது **mountpoint**-ல் **mount** ஆகி விடுகின்றன. இவை /etc/fstab எனும் **file** மூலம் குறிப்பிடப்படுகின்றன.

இதில் 6 **column** களும் பல வரிகளும் உள்ளன. ஒவ்வொரு வரியும் ஒரு **filesystem**க்கானது.

```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~
File Edit View Search Terminal Help
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~$ cat /etc/fstab
# /etc/fstab: static file system information.
#
# Use 'blkid' to print the universally unique identifier for a
# device; this may be used with UUID= as a more robust way to name devices
# that works even if disks are added and removed. See fstab(5).
#
# <file system> <mount point>   <type>  <options>       <dump>  <pass>
proc            /proc           proc    nodev,noexec,nosuid 0       0
# / was on /dev/sda6 during installation
UUID=42ce71a3-3768-4b1d-84d0-dc96907bb5fa /               ext4    errors=remount-ro 0
1
# /home was on /dev/sda7 during installation
UUID=056fecab-bfed-4897-a90e-ef648a96361e /home           ext4    defaults        0       2
# swap was on /dev/sda5 during installation
UUID=4a51d987-57fd-4246-a841-821dcdf59a84 none            swap    sw              0       0
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~$
```

Filesystem வகை:

இதில் பயன்படும் பொதுவான filesystem வகைகள்:

ext3/ext4 : குனூ / லினக்சின் பொதுவான file system

iso9660 : iso file

proc : proc எனும் கெர்னலுக்கு பயன்படும் virtual file system

vfat : fat 16, fat 32 file system

auto : filesystem வகையை தானாக கண்டறிதல்

nts : லினக்ஸ் network file system

smbfs : விண்டோஸ் network file system

Mount Options:

Mount கட்டளையுடன் -o சேர்த்து , உடன் பயன்படுத்தலாம்.

/etc/fstab-லும் குறிப்பிடலாம்.

noauto : தானாகவே boot ஆகும் போது mount ஆவதை

தவிர்க்கிறது. cd/dvd போன்றவற்றுக்கு பயன்படும்.

ro : Read-Only ஆக mount செய்தல்

users : root அல்லாத சாதாரண பயனர்களும் mount (அ) unmount செய்யலாம்.

user : சாதாரண பயனர், தானே mount செய்தவற்றை மட்டுமே unmount செய்தல்

/etc/fstab-ல் உள்ள 5 வது column, dump எனப்படும்.

இது dump, restore போன்ற backup செயல்களுக்கு பயன்படும். சாதாரண filesystem களுக்கு 1 எனவும், cd/dvd/usb போன்ற removable media-க்கு 0 எனவும்

தரவும்.

6 வது column-ஆனது pass-no எனப்படும். boot ஆகும் போது file system check நடைபெற வேண்டுமா என்பதை குறிக்கிறது.

root / க்கு 1

boot-ன் போது mount ஆகாத file system-களுக்கு 0

பிற file system-களுக்கு 2

ஒரு file-ஐ mount செய்தல்:

iso file போன்ற file-களை ஒரு device போலவே mount செய்யலாம்.

1.44 MB உள்ள ஒரு file ஐ உருவாக்குவோம்.

```
dd if=/dev/zero of=disk.img bs=1024 count=1400
```

dd = disk dump

if = input file

of = output file

bs = block size

count = count of dump operation

இதன் மூலம் பெரிய அளவிலான file-களை மிக எளிதாக உருவாக்கலாம்.

file system களை back levelல் copy செய்யவும் இது பயன்படும்.

```
dd if=/dev/sdb1 of=usb.img
```

```
mount -o loop usb.img /mnt/disk
```

மேலும் சில disk தொடர்பான கட்டளைகள்:

disk space பயன்பாடு:

df

```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~
File Edit View Search Terminal Help
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~$ df
Filesystem     1K-blocks     Used Available Use% Mounted on
/dev/sda6       29223512  4325936  23432796  16% /
udev             1019288        4   1019284   1% /dev
tmpfs             410624      868    409756   1% /run
none                5120        0      5120   0% /run/lock
none             1026552      256   1026296   1% /run/shm
/dev/sda7       38973500 22263060  14757368  61% /home
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~$
```

இது disk space-ன் அளவு, பயன்பாடு மீதம் உள்ளது. Mount point இவற்றை தரும்.

df -h

```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~
File Edit View Search Terminal Help
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~$ df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda6        28G  4.2G   23G  16% /
udev            996M  4.0K  996M   1% /dev
tmpfs           401M  868K  401M   1% /run
none            5.0M     0  5.0M   0% /run/lock
none           1003M  256K 1003M   1% /run/shm
/dev/sda7        38G   22G   15G  61% /home
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~$
```

-h ஆனது human readable மனிதருக்கு புரியும் வகையில் MB, GB ல் தரும்.

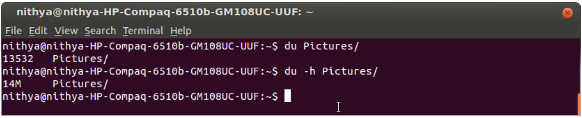
folder-ன் அளவு:

du <foldername>

ஒரு folder-ன் அளவை தருகிறது.

du -h

du -sh



file system-ஐ repair செய்தல்:

தொடர்ந்த இயக்கம், அடிக்கடி திடீரென இயக்கம் நிறுத்துதல், மின்தடை போன்ற காரணங்களால், நமது filesystem பழுதடையலாம்.

Hardware – வன்பொருள் பழுதாகாமல் இருக்கும் வரையில் இதை சரிசெய்ய வாய்ப்பு உள்ளது. சில நேரங்களில் தகவல் இழப்பும் ஏற்படலாம்.

```
fsck <option> <device>
```

```
sudo fsck /dev/sda5
```

ஏதேனும் பிழை சரிசெய்யப்படும் எனில் உங்களிடம் அனுமதி கேட்கப்படும்.

```
sudo fsck -y -f /dev/sda5
```

-y = yes for questions

-f = force

fdisk இயக்கும் முன் file system ஐ unmount செய்ய வேண்டும்.

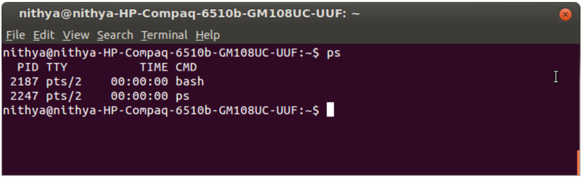
Process Management

நமது கணினியில் இயங்கிக் கொண்டிருக்கும் ஒவ்வொரு சின்னச் சின்ன விஷயமும் ஒரு process எனப்படும். இது parent process மற்றும் child process என இரு வகைப்படும். இவை அனைத்திற்கும் ஆதிமூலமாக init எனும் process அமையும். ஒவ்வொரு process-ம் ஒரு pid மூலம் refer செய்யப்படும்.

ps command

இந்த ps command-ஆனது தற்சமயம் நமது கணினியில் ஓடிக்கொண்டிருக்கும் அனைத்து process-ஐயும் வெளிப்படுத்த உதவும்.

```
ps
```



ps ax, ps -aux, ps -ef, ps -l

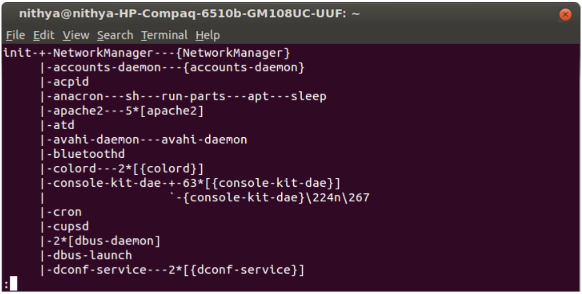
போன்றவை இன்னும் விரிவான தகவல்களை தருகின்றன.

pstree command

pstree command- ஆனது நமது கணினியில் இயங்கிக் கொண்டிருக்கும் அனைத்து process-ஐயும், parent process மற்றும் child process- ஐ அடிப்படையாகக் கொண்டு tree

வடிவில் அமைத்து வெளிப்படுத்தும்.

pstree



```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~
File Edit View Search Terminal Help
init-+-NetworkManager---{NetworkManager}
     |-accounts-daemon---{accounts-daemon}
     |-acpid
     |-anacron---sh---run-parts---apt---sleep
     |-apache2---5*[apache2]
     |-atd
     |-avahi-daemon---avahi-daemon
     |-bluetoothd
     |-colord---2*[{colord}]
     |-console-kit-dae-+-63*[{console-kit-dae}]
     |                 `-{console-kit-dae}\224n\267
     |-cron
     |-cupsd
     |-2*[dbus-daemon]
     |-dbus-launch
     |-dconf-service---2*[{dconf-service}]
:
```

top command

top command- ஆனது ஒரு process எடுத்துக்கொள்ளும் நேரத்தின் அடிப்படையில் அனைத்து process-ஐயும் இறங்கு வரிசையில் வெளிப்படுத்தும். இதன் output-ஆனது real time monitoring மூலம் மாறிக்கொண்டே இருப்பதை நாம் நேரடியாகக் காணலாம்.

```
$ top
```

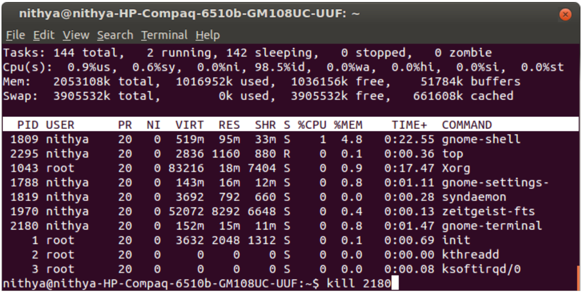
```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~
File Edit View Search Terminal Help
top - 09:05:21 up 11 min,  1 user,  load average: 0.47, 0.55, 0.41
Tasks: 144 total,   1 running, 143 sleeping,   0 stopped,   0 zombie
Cpu(s):  0.8%us,  0.5%sy,  0.0%ni, 97.8%id,  0.8%wa,  0.0%hi,  0.0%si,  0.0%st
Mem:   2053108k total,  1015664k used,  1037444k free,    51644k buffers
Swap:  3905532k total,        0k used,  3905532k free,   659900k cached

  PID USER      PR  NI  VIRT  RES  SHR S %CPU %MEM    TIME+  COMMAND
 1043 root      20   0 83500  18m 7404 S    1  0.9   0:15.94 Xorg
 1809 nithya    20   0  519m  95m  33m S    1  4.8   0:20.67 gnome-shell
 2295 nithya    20   0  2836 1160  880 R    1  0.1   0:00.05 top
 2180 nithya    20   0  152m  14m  11m S    0  0.7   0:01.26 gnome-terminal
    1 root      20   0  3632 2048 1312 S    0  0.1   0:00.69 init
    2 root      20   0     0    0    0 S    0  0.0   0:00.00 kthreadd
    3 root      20   0     0    0    0 S    0  0.0   0:00.07 ksoftirqd/0
    5 root      20   0     0    0    0 S    0  0.0   0:00.36 kworker/u:0
    6 root      RT   0     0    0    0 S    0  0.0   0:00.00 migration/0
    7 root      RT   0     0    0    0 S    0  0.0   0:00.00 watchdog/0
```

Kill command

இது ஓர் இயங்கிக் கொண்டிருக்கும் **process**-ஐ நிறுத்தப் பயன்படும். அது மட்டுமல்லாமல் ஒரு **process**-க்கு பல்வேறு வகையான **signal**-களை அனுப்பவும் பயன்படுகிறது

$ **kill** 2180

பின்வரும் **screenshot**-ல் நமது கணிணியில் இருக்கும் **terminal**-ன் **pid** 2180 என்பதைக் காணலாம். எனவே **kill** 2180 எனக் கொடுக்கும் போது, நமது **terminal** நிறுத்தப்பட்டு மூடப்படுவதைக் காணலாம்.



killall <program name>

என இயக்கி, ஒரு program ன் பெயரைக் கொண்டு, அதை kill செய்யலாம்.

உபுண்டு லினக்சில் மென்பொருட்களை நிறுவுதல்

மென்பொருட்களை நிறுவ synaptic, software center போன்ற GUI மென்பொருட்கள் இருந்தாலும், அவற்றுக்கான command-களை அறிவது முக்கியம்.

Apt

பொதுவாக, லினக்சில் எல்லா மென்பொருட்களும், பல்வேறு தனித்தனி கூறுகளாகவே உருவாக்கப்படுகின்றன. அவை user interface, module, library, translation என பல பகுதிகளாகவே நிறுவப்படுகின்றன. இந்த பல்வேறு பகுதிகளும் ஒன்றையொன்று சார்ந்தவை. இதை dependancy என்பர்.

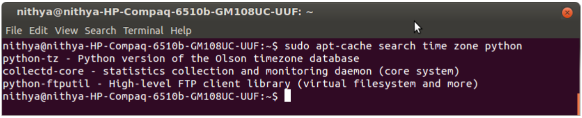
Package management system என்பது மென்பொருட்களை அவற்றின் சார்பு மென்பொருட்களோடு நிறுவுதல், மேம்படுத்துதல், நீக்குதல் போன்றவற்றை செய்கின்றது. Redhat சார்ந்த distribution-களில் yum என்பதும், debian,

ubuntu களில் APT, Arch linux-ல் pacman ஆகியவை இந்த வேலைகளை செய்கின்றன.

மென்பொருட்களை தேடுதல்:

sudo apt-cache search [search term 1] [search term 2] … [search term n]

ஏதேனும் ஒரு மென்பொருளை தேட



repositories:

debian, ubuntu, centos, suse போன்ற distributionகளில் எல்லா மென்பொருட்களும் இணையத்தில் அவற்றில் சர்வர்களில் package செய்யப்பட்டு வைக்கப்படுகின்றன. இது repository எனப்படும்.

repository-ல் இருந்து எளிதாக மென்பொருட்களை தேடி

நிறுவலாம். புதிய பதிப்புகள் வரும்போது, எளிதாக மேம்படுத்தலாம்.

debian, ubuntu-ல் /etc/apt/sources.listல் இந்த repository களை சேர்க்கலாம்.

புதிதாக ஒரு repo-ஐ சேர்த்தல்

sudo add-apt-repository [repository name here]

```
sudo add-apt-repository ppa:tualatrix/ppa
```

Source-ஐ மேம்படுத்துதல்:

புது repo-ஐ சேர்த்த பின் repo தகவல்களை மேம்படுத்த வேண்டும்.

```
sudo apt-get update
```

```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~
File Edit View Search Terminal Help
Ign http://extras.ubuntu.com precise/main Translation-en_IN
Ign http://extras.ubuntu.com precise/main Translation-en
Get:8 http://security.ubuntu.com precise-security/restricted Sources [2,494 B]
Get:9 http://security.ubuntu.com precise-security/universe Sources [28.1 kB]
Get:10 http://security.ubuntu.com precise-security/multiverse Sources [1,804 B]
Get:11 http://security.ubuntu.com precise-security/main i386 Packages [342 kB]
Get:12 http://security.ubuntu.com precise-security/restricted i386 Packages [4,620 B]
Get:13 http://security.ubuntu.com precise-security/universe i386 Packages [86.0 kB]
Get:14 http://security.ubuntu.com precise-security/multiverse i386 Packages [2,640 B]
Get:15 http://security.ubuntu.com precise-security/main TranslationIndex [74 B]
Get:16 http://security.ubuntu.com precise-security/multiverse TranslationIndex [72 B]
Get:17 http://security.ubuntu.com precise-security/restricted TranslationIndex [72 B]
Get:18 http://in.archive.ubuntu.com precise-updates/restricted Sources [7,006 B]
Get:19 http://security.ubuntu.com precise-security/universe TranslationIndex [73 B]
Get:20 http://security.ubuntu.com precise-security/main Translation-en [155 kB]
Get:21 http://in.archive.ubuntu.com precise-updates/universe Sources [96.2 kB]
Hit http://security.ubuntu.com precise-security/multiverse Translation-en
Hit http://security.ubuntu.com precise-security/restricted Translation-en
Get:22 http://security.ubuntu.com precise-security/universe Translation-en [52.4 kB]
Get:23 http://in.archive.ubuntu.com precise-updates/multiverse Sources [8,343 B]
Get:24 http://in.archive.ubuntu.com precise-updates/main i386 Packages [714 kB]
71% [24 Packages 96.9 kB/714 kB 14%]                              19.9 kB/s 31s
```

நிறுவுதல்:

புது மென்பொருளை நிறுவுதல்:

sudo apt-get install [package name 1] [package name 2] ... [package name n]

இது தரப்பட்ட மென்பொருளையும், அதற்கு துணையான பிற மென்பொருட்கள் அனைத்தையும் repository-ல் இருந்து தரவிறக்கம் செய்து நிறுவுகிறது.

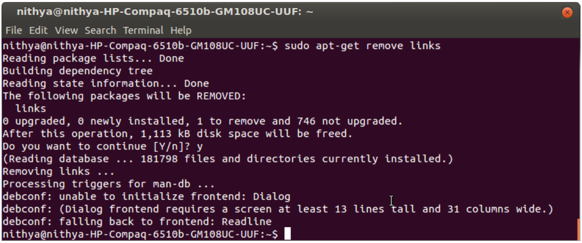
```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~
File Edit View Search Terminal Help
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~$ sudo apt-get install rhythmbox
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  gir1.2-rb-3.0 librhythmbox-core5 rhythmbox-data rhythmbox-mozilla
  rhythmbox-plugin-cdrecorder rhythmbox-plugin-magnatune rhythmbox-plugins
The following packages will be upgraded:
  gir1.2-rb-3.0 librhythmbox-core5 rhythmbox rhythmbox-data rhythmbox-mozilla
  rhythmbox-plugin-cdrecorder rhythmbox-plugin-magnatune rhythmbox-plugins
8 upgraded, 0 newly installed, 0 to remove and 738 not upgraded.
Need to get 1,574 kB of archives.
After this operation, 4,096 B disk space will be freed.
Do you want to continue [Y/n]?
```

மென்பொருளை நீக்குதல்:

ஏதேனும் ஒரு மென்பொருளையும் அதன் சார்பு

மென்பொருட்களையும் நீக்க

sudo apt-get remove [package name 1] [package name 2] ... [package name n]



மென்பொருட்களின் configuration file மற்றும் அவற்றின் directory களையும் நீக்க –purge பயன்படுத்தலாம்.

sudo apt-get remove –purge <name1> <name2>

மேம்பாடு – upgrade:

ஏதேனும் ஒரு மென்பொருளை மட்டும் மேம்படுத்த

sudo apt-get upgrade <name1> <name2> … <name n>

எல்லா மென்பொருட்களையும் upgrade செய்ய

```
sudo apt-get upgrade
```

இது மேம்பாடு செய்யும் மென்பொருட்கள் காட்டி, அனுமதி பெற்று, பின் மேம்பாடு செய்யும்.

```
sudo apt-get -s upgrade
```

இது மேம்பாடு செய்யும் செயலை நடத்தாமல், simulate

மட்டும் செய்யும்.

சுத்தம் செய்தல்:

repo-லிருந்து பதிவிறக்கம் செய்த பழைய deb file-கள் இடத்தை அடைத்துக் கொண்டு இருக்கும். அவற்றை நீக்க

```
sudo apt-get clean
```

நிறுவப்பட்ட மென்பொருள் பட்டியல்:

```
sudo dpkg –list
```

இவை நிறுவப்பட்ட மென்பொருட்கள் பட்டியலையும், குறிப்பிட்ட மென்பொருளை தேடவும் பயன்படுகின்றன.

Deb file ஐ நிறுவுதல்:

debian/ubuntu மென்பொருட்கள் packages deb என்ற அமைப்பில் இருக்கும். இவை தனியாக கிடைத்தால் அல்லது நாமே உருவக்கினால், பின்வரும் கட்டளை மூலம் நிறுவலாம்.

```
sudo dpkg -i <software.deb>
```

Shell Programming பற்றிய அறிமுகம்

இந்தப் பகுதியில் நாம், shell-ன் மூலம் வழங்கப்படுகின்ற ஒருசில அம்சங்களைப் பயன்படுத்தி, எவ்வாறு ஒரு program-ஐ எழுதி இயக்குவது என்று காணலாம்.

மேலும் shell-ம் ஒரு programming language-ஐப் போலவே செயல்படுவதற்கான அனைத்துத் தகுதிகளையும் கொண்டுள்ளது. அதாவது programming language-ல் உள்ள programming constructs, logical மற்றும் conditional operators, command substitution, escape mechanisms,

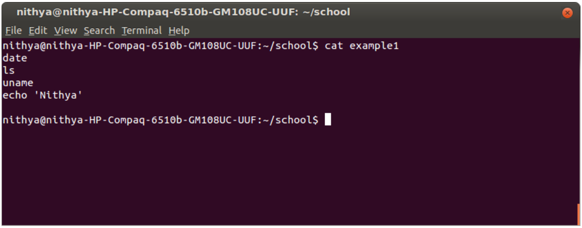
positional parameters என்பது போன்ற அனைத்து வகையான சிறப்பு அம்சங்களையும் shell-ஆனது உள்ளடக்கியுள்ளது.

Shell Script-க்கான விளக்கம்

இதுவரை நாம் unix-ல் உள்ள பல்வேறு வகையான commands-ஐப் பற்றிப் பார்த்துள்ளோம். இத்தகைய commands சிலவற்றை நம் தேவைக்கு ஏற்றவாறு ஒன்றாகத் தொகுத்து அந்தத் தொகுப்புக்கு ஒரு பொதுவான பெயர் வைத்தால் அதுவே “shell script” எனப்படும்.

உதாரணத்துக்கு date, ls, uname, echo எனும் நான்கு தனித்தனி unix commands-ஐ ‘example1′ எனும் file-க்குள்

போடவும். இது பின்வருமாறு.



இங்கு example1 எனும் ஓர் புதிய shell script உருவாக்கப்பட்டுள்ளது. அடுத்ததாக இந்த script-ஐ எவ்வாறு இயக்கி, அதன் output-ஐப் பெறுவது என்று பார்க்கலாம்.

Shell Script-ஐ இயக்குதல்

ஒரு shell script-ஐ நாம் பின்வரும் இரண்டு வழிகளில் இயக்கலாம்.

வழி 1: sh எனும் unix command-க்கு தமது script file-ஐ argument-ஆகக் கொடுப்பதன் மூலம் ஒரு shell script-ஐ இயக்கலாம். இது பின்வருமாறு.

```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~/school
File  Edit  View  Search  Terminal  Help
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~/school$ sh example1
Sun Apr  7 22:17:48 IST 2013
abc  animals  Companies  example1  fruits  introduction  language1  tamil  xyz
Linux
Nithya
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~/school$
```

இங்கு example1 எனும் shell script-ஆனது அதனுள் அடங்கியிருக்கும் பல்வேறு தனித்தனி unix commands-ன் output-ஐ ஒன்றாக வெளிப்படுத்துவதைக் காணலாம்.

வழி 2: அடுத்ததாக chmod எனும் unix command மூலம் ஒரு file இயக்கப்படுவதற்குத் தேவையான execute permission-ஐ அளித்தபின்னர் அந்த script file-ஐ நேரடியாகவே நாம் execute செய்யலாம். இது பின்வருமாறு.

Variables

Unix-ல், நாம் பயன்படுத்தும் variables-ஐ, local variables மற்றும் global variable என்று இரண்டு வகையாகப் பிரிக்கலாம். அவற்றைப் பற்றி இங்கு விளக்கமாகக் காண்போம்.

Global Variables

Global Variables என்பது நாம் இயங்கிக்கொண்டிருக்கும் unix இயங்குதளத்தில் எங்கு பயன்படுத்தினாலும், அதற்கென்று வரையறுக்கப்பட்ட மதிப்பினை மட்டும் வெளிப்படுத்தும். எனவே இவை environment variables எனவும் அழைக்கப்படும்.

உதாரணத்துக்கு PATH எனும் variable-ல் ஒருசில directory-களின் விவரங்கள் சேமித்துவைக்கப்பட்டுள்ளன. அவ்வாறே PS1 எனும் variable-ல் shell prompt-ன் விவரங்களும், HOME எனும் variable-ல் நாம் பயன்படுத்தும் home directory-ன் விவரங்களும் சேமித்து வைக்கப்பட்டுள்ளன.

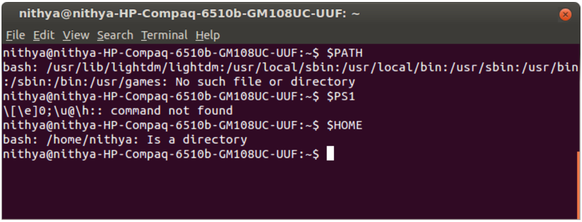
$ (dollar) எனும் குறியுடன் சேர்த்து இத்தகைய variables-ஐ நாம் அணுகும்போது, அவை அவற்றின் மதிப்பினை

வெளிப்படுத்துவதைக் காணலாம்.

$ $PATH

$ $PS1

$ $HOME

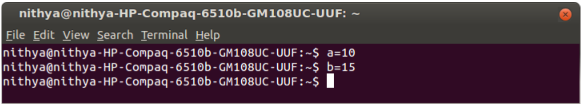


Local Variables

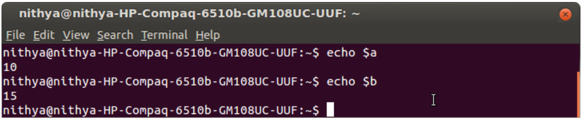
ஒரு shell environment-ல் வரையறுக்கப்படும் variable-ஆனது local variable எனப்படும். இவை shell environment-ஐத் தவிர வேறு எங்கு பயன்படுத்தினாலும், அதற்குரிய மதிப்பினைத் தராது. எனவே இவை ‘shell variables’ எனவும் அழைக்கப்படும்.

ஒரு local variable -ஐ உருவாக்குதல்:

உதாரணத்துக்கு a,b எனும் 2 variable-ன் மதிப்பை shell environment-ல் பின்வருமாறு வரையறுக்கவும்.



பின்பு இந்த variables-ஐ echo செய்து பார்க்கவும். இவை அதனதன் மதிப்புகளை வெளிப்படுத்துவதைக் காணலாம்.



அடுத்தபடியாக இந்த variables-ஐ உள்ளடக்கிய shell script-ஐ உருவாக்கி, அதனை இயக்கிப் பார்க்கவும்.

இப்போது அந்த variables செயல்படாது. ஏனெனில் அவை local variables அதாவது shell environment-ல் மட்டுமே இயங்கக்கூடிய shell variables.

இவற்றை நாம் global variables-ஆக மாற்றினால் மட்டுமே, இதனை நாம் எங்கு வேண்டுமானாலும் பயன்படுத்த முடியும்.

மேலும் நமது environment-ல் என்னென்ன shell variables

உள்ளன என்பதைத் தெரிந்துகொள்ள set எனும் command-ஐப் பயன்படுத்தலாம்.

```
$ set
```

Local variables-ஐ Global variables-ஆக மாற்றுதல்

ஒரு local variable-ஐ global variable-ஆக மாற்றுவதற்கு export command பயன்படும்.

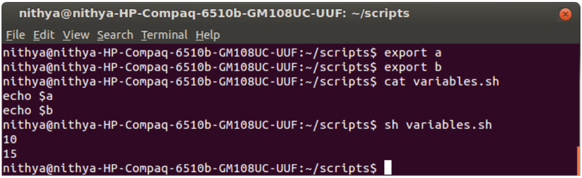
உதாரணத்துக்கு நாம் உருவாக்கிய a,b எனும் 2 local variable-ஐ global variable-ஆக மாற்றுவதற்கு, shell environment-ல் பின்வருமாறு export command-ஐ

அமைக்கவும்.

$ export a

$ export b

பின்னர் இந்த variables-ஐ உள்ளடக்கிய shell script-ஐ இயக்கிப் பார்க்கவும். இவை அதனதன் மதிப்புகளை வெளிப்படுத்துவதைக் காணலாம்.



Special Variables

Local variable மற்றும் global variable அல்லாது unix-ல் பயன்படுத்தப்படும் ஒருசில சிறப்புவகை variable-ஐப் பற்றி இங்கு காணலாம்.

$# இது எத்தனை positional parameters உள்ளது எனும் எண்ணிக்கையைக் கொடுக்கும்.

$- இது shell options-ஐக் குறிக்கும்,

$? கடைசியாக execute செய்யப்பட்ட command-ன் exit நிலையைக் காட்டும்.

$$ தற்போதுள்ள shell-ன் process எண்ணைக் காட்டும்.

$! கடைசி background command-ன் process எண்ணைக் காட்டும்.

$0 தற்போது execute செய்துகொண்டிருக்கும் command-ன் பெயரைக் குறிக்கும்.

$* Positional parameters-ன் பட்டியலைக் காட்டும்.

Shell Script- 2

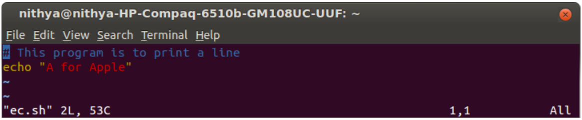
Hash # குறியின் பயன்பாடு

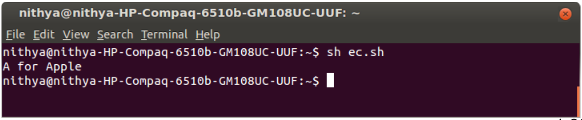
ஒரு வரியின் தொடக்கத்தில் # எனும் குறியை முதலில் type செய்துவிட்டு, அதைத் தொடர்ந்து நாம் எதை type செய்தாலும், அந்த வரி execution நேரத்தில் கணக்கில் எடுத்துக் கொள்ளப்படாது.

உதாரணத்துக்கு நாம் எழுதியுள்ள shell script என்ன செய்கிறது என்பதை ஒரு சில வரிகளால் நாம் விளக்க

விரும்பினால், இந்த # குறியைப் பயன்படுத்தி, அந்த வரிகளை shell script-லேயே எழுதிவிடலாம்.

உதாரணத்துக்கு ec.sh எனும் file-ல் பின்வருமாரு ஒரு comment-ஐ அடித்துவிட்டு, அதனை run செய்து பார்க்கவும்.

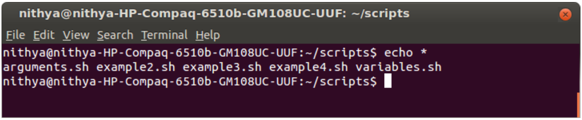




இது வெளிப்படுத்தியுள்ள output-ல் நாம் comment-ஆக கொடுத்த வரி வெளிப்படாமல் இருப்பதை கவனிக்கவும்.

Escape Mechanism

இதை நீங்கள் பின்வரும் உதாரணத்தின் மூலம் தெளிவாகப் புரிந்து கொள்ளலாம். ✱ (Asterisk symbol) என்பதை echo செய்து பார்க்கவும்.

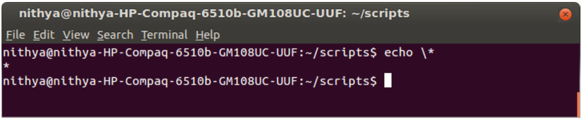


இங்கு echo command-ஆல் ✱-ஐ அப்படியே

வெளிப்படுத்த முடியவில்லை. ஏனெனில் * என்பது unix-ஐப் பொருத்தவரை ஒரு special character.

அதாவது * எனக் குறிப்பிட்டால் என்ன செய்ய வேண்டுமோ, அதைத்தான் Unix செய்யும். இதையும் ஒரு சாதாரண character-ஆக எண்ணி அப்படியே வெளிப்படுத்த unix-க்குத் தெரியாது.

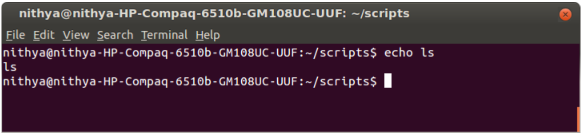
எனவே இதுபோன்ற special characters-ஐ சாதாரண character-ஆகக் கருதுமாறு unix-க்கு நீங்கள் சொல்ல விரும்பினால், இத்தகைய character-க்கு முன்னால் backslash (\)குறியை இட வேண்டும்.



இப்போது ✱ எனும் குறி அப்படியே வெளிப்படுத்தப்பட்டிருப்பதைக் காணலாம். இதுவே escape mechanism எனப்படும்.

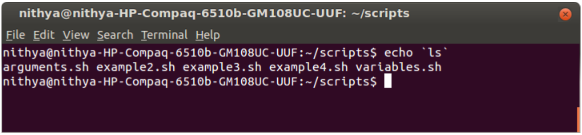
Command substitution

இதையும் நீங்கள் பின்வரும் உதாரணத்தின் மூலம் தெளிவாகப் புரிந்து கொள்ளலாம்.முதலில் ஏதாவதொரு unix command-ஐ echo செய்து பார்க்கவும்.



இங்கு ls எனும் வார்த்தை அப்படியே வெளிப்படுதப்படுவதைக் காணலாம்.

அடுத்ததாக இந்த ls command-ன் இருபுறங்களிலும் backticks-ஐ இணைத்து echo செய்து பார்க்கவும்.



இப்போது ls என்பது ஒரு command-ஆகக் கருதப்பட்டு அதன் output-ஆனது echo செய்யப்படுவதைக் காணலாம்.

இதுவே command substitution எனப்படும்.

Positional Parameters

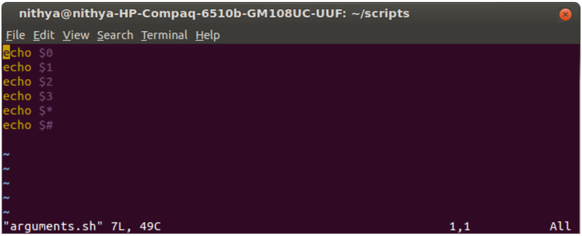
Shell script-ஆனது command line-ல் இருந்து arguments-ஐ நேரடியாகவும் பெற்றுக்கொள்ளும். அப்போது shell script-க்குள் கொடுக்கப்படும் $1 என்பது முதல் argument-ஐயும், $2 என்பது இரண்டாவது argument-ஐயும், $3 என்பது மூன்றாவது argument-ஐயும் குறிக்கும்.

மேலும் $0 என்பது shell script-ன் பெயரையும், அவ்வாறே

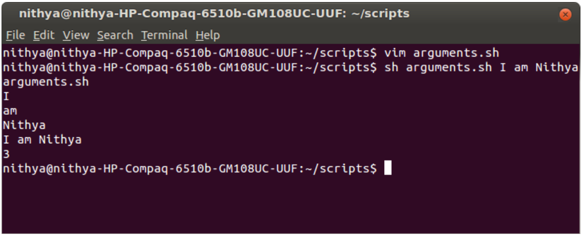
$✶ என்பது command line-ல் கொடுக்கப்பட்டுள்ள மொத்த arguments-ன் எண்ணிக்கையையும் குறிக்கும்.

இவ்வாறாக $1,$2,$3 என்பதெல்லாம், arguments-ன் position-ஐப் பொறுத்து மதிப்புகளை ஏற்றுக்கொள்வதால், இவை positional parameters எனப்படுகின்றன.

உதாரணத்துக்கு arguments.sh எனும் shell script-க்குள் பின்வருமாறு அடிக்கவும்.



பின்னர் arguments.sh-ஐ command line-லேயே arguments-ஐக் கொடுத்து பின்வருமாறு இயக்கவும்.



```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~/scripts
File Edit View Search Terminal Help
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~/scripts$ vim arguments.sh
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~/scripts$ sh arguments.sh I am Nithya
arguments.sh
I
am
Nithya
I am Nithya
3
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~/scripts$
```

இங்கு நாம் கொடுத்த arguments அனைத்தும், shell script-க்குள் உள்ள positional parameters-ஐப் பொறுத்து output-ஆக வெளிப்படுவதைக் காணலாம்.

File descriptors

File descriptors எனப்படும் 0,1 மற்றும் 2 எனும் எண்கள்

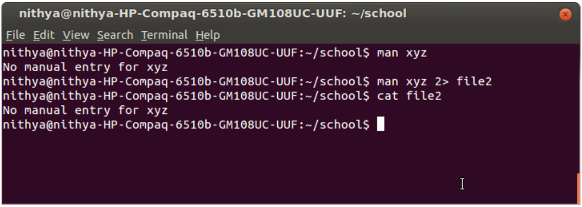
முறையே standard input, standard output மற்றும் standard error-ஐக் குறிக்கின்றன.

ஒரு shell script-ஆனது இயக்கப்படும்போது, இடையில் ஒருசில commands சரிவர இயங்காமல், error messages திரையில் வெளிப்படுத்தப்பட்டு, பாதியிலேயே அந்த program தடைபட வாய்ப்பு உள்ளது.

இதுபோன்ற descriptors-ஐப் பயன்படுத்துவதன் மூலம், நாம் error messages அனைத்தையும் திரையில் வெளிப்படுத்த விடாமல், ஒரு file-க்குள் செலுத்தி, அந்த program பாதியிலேயே தடைபடுவதிலிருந்து காக்கலாம்.

உதாரணத்துக்கு பின்வரும் man xyz எனும் command கண்டிப்பாக திரையில் error-ஐ வெளிப்படுத்தும். இதனை error descriptor-க்கான எண் 2-ன் மூலம் பின்வருமாறு ஒரு file-க்குள் செலுத்தலாம்.

```
$ man xyz 2> file2
```



அதே போன்று man ls xyz எனும் command, ls-க்கு output-ஐயும் , xyz-க்கு error-ஐயும் வெளிப்படுத்தும். இதுபோன்ற சமயங்களில், output-ஐ ஒரு file-லிலும், error-ஐ மற்றொரு file-லிலும் பின்வருமாறு சேமிக்கலாம்.

```
$ man ls xyz 1> opfile 2> errfile
```

```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~/school
File Edit View Search Terminal Help

nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~/school$ man ls xyz
No manual entry for xyz
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~/school$ man ls xyz 1> opfile 2> errfile
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~/school$ cat opfile
```

```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~/school
File Edit View Search Terminal Help
       This is free software: you are free to change and redistribute it.  There is
       NO WARRANTY, to the extent permitted by law.

SEE ALSO
       The full documentation for ls is maintained as a  Texinfo  manual.   If  the
       info and ls programs are properly installed at your site, the command

              info coreutils 'ls invocation'

       should give you access to the complete manual.



GNU coreutils 8.12.197-032bb       September 2011                           LS(1)
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~/school$ cat errfile
No manual entry for xyz
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~/school$
```

அடுத்ததாக output மற்றும் error இரண்டையும் ஒரே file-ல் சேமிக்க command-ஐ பின்வருமாறு அமைக்கவும்.

$ man ls xyz > operrfile 2>&1

```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~/school
File Edit View Search Terminal Help
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~/school$ man ls xyz > operrfile 2>&1
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~/school$ cat operrfile
```

```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~/school
File Edit View Search Terminal Help
       version 3 or later <http://gnu.org/licenses/gpl.html>.
       This is free software: you are free to change and redistribute it.  There is
       NO WARRANTY, to the extent permitted by law.

SEE ALSO
       The full documentation for ls is maintained as a  Texinfo  manual.   If  the
       info and ls programs are properly installed at your site, the command

              info coreutils 'ls invocation'

       should give you access to the complete manual.



GNU coreutils 8.12.197-032bb       September 2011                          LS(1)
No manual entry for xyz
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~/school$
```

Shift Command

ஒரு shell script-க்குள் நாம் அதிகபட்சம் $1 லிருந்து $9 வரை positional parameters-ஐப் பயன்படுத்தலாம். இதற்கு மேலும் நாம் பயன்படுத்த விரும்பினால், shift command-ன் உதவியை நாட வேண்டும்.

exit Command

exit command-ஆனது ஓர் இயங்கிக் கொண்டிருக்கும் shell script-ஐ நிறுத்த உதவும். அந்த shell script-ஆனது வெற்றிகரமாக இயங்கி முடித்த பின்னர் நிறுத்தப்பட்டதெனில்

0-வும், இல்லையெனில் 1-ம் வெளிப்படும்.

Numerical Comparisons

ஒரு **shell script**-க்குள் இரண்டு எண்களை எவ்வாறு ஒப்பீடு செய்வது என்று இங்கு பார்க்கலாம். உதாரணத்துக்கு, 3 மற்றும் 5 எனும் எண்கள் எவ்வாறெல்லாம் ஒப்பீடு செய்யப்படலாம் என்பது பின்வருமாறு கொடுக்கப்பட்டுள்ளது.

3 -**eq** 5 (3 மற்றும் 5 சமமாக (**equals**) உள்ளதா என்பதை

சரிபார்க்கும்)

3 -**ne** 5 (3 மற்றும் 5 சமமாக இல்லையா (**not equals**)

என்பதை சரிபார்க்கும்)

3 -**gt** 5 (3-ஆனது 5-ஐ விடப் பெரியதா (**greater than**)

என்பதை சரிபார்க்கும்)

3 -**lt** 5 (3-ஆனது 5-ஐ விடச் சிறியதா (**greater than**)

என்பதை சரிபார்க்கும்)

3 -**ge** 5 (3-ஆனது 5-க்குச் சமமாகவோ அல்லது அதை விடப் பெறிய எண்ணாகவோ (**greater than or equal to**) உள்ளதா என்பதை சரிபார்க்கும்)

3 -**le** 5 (3-ஆனது 5-க்குச் சமமாகவோ அல்லது அதை விடச் சிறிய எண்ணாகவோ (**greater than or equal to**) உள்ளதா என்பதை சரிபார்க்கும்)

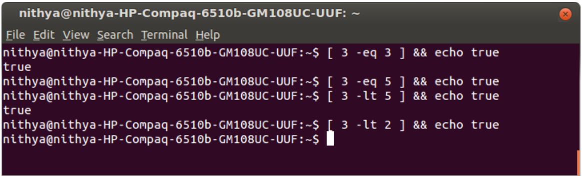
இதைத் தற்சமயம் **command line**-ல் நேரடியாகச் செய்து பார்க்கலாம்.

$ [3 -eq 3] && echo true

$ [3 -eq 5] && echo true

$ [3 -lt 5] && echo true

$ [3 -lt 2] && echo true



இங்கு முதல் **command**-ஆன 3 -**eq** 3 என்பது சரி என்பதால் **true** என்பதை வெளிப்படுத்துகிறது. அடுத்ததாக 3 -**eq** 5 என்பது தவறு என்பதால் **true** என்பது வெளிப்படவில்லை. அவ்வாறே அடுத்தடுத்த **commands**-ம் செயல்பட்டுள்ளதை கவனிக்கவும்.

Arithmetic Operators & Expressions

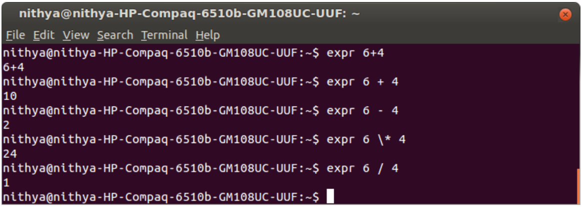
கூட்டல், கழித்தல், பெருக்கல், வகுத்தல் என்பதற்குப் பயன்படும் +, -, *, / ஆகிய நான்கும் **arithmetic operators** எனப்படும். இத்தகைய கணிதக் குறியீடுகளை நாம் எழுதும் **shell script**-ல் பயன்படுத்தி ஒரு சில எளிய கணக்குகளைப்

போடலாம். இது பின்வருமாறு.

```
$ expr 6 + 4
$ expr 6 – 4
$ expr 6 \* 4
$ expr 6 / 4
```

இங்கு பெருக்கலைக் குறிக்க * **operators**-ஐப் பயன்படுத்தும் போது, மறக்காமல் \ எனும் **escape operator**-ஐப் பயன்படுத்தவும்.

அதேபோல், / operator-ஐப் பயன்படுத்தி வகுத்தல் கணக்கைப் போடும்போது, அதன் விடை தசம எண்ணாக வெளிப்படாமல், அதற்கு சற்று நிகரான முழு எண்ணாக வெளிப்பட்டிருப்பதைக் கவனிக்கவும்.



Conditional Operators

சில சமயம் நாம் ஒரு சிலcommands-ன் successful execution-ஐப் பொறுத்தோ அல்லது failed execution-ஐப் பொறுத்தோ வேறொருசில commands-ஐ execute செய்ய நேரிடும். எனவே இதற்குப் பயன்படும் && மற்றும் || ஆகியவை conditional execution operators எனப்படும்.

&& எனும் operator-ஆனது successful execution-ஐப் பொறுத்தும்,

|| எனும் operator-ஆனது failed execution-ஐப் பொறுத்தும் வேறொரு சில commands-ஐ execute செய்யும்.

$ grep Apple fruits && echo “found” || echo “Not found”

$ grep Mango fruits && echo “found” || echo “Not found”

Shell Scripting – 3

Programming language Constructs

Shell-ம் மற்ற programming language-ஐப் போலவே looping & branching, operations on variables, file creation மற்றும் argument passing போன்ற பல்வேறு வகையான சிறப்பு அம்சங்களை உள்ளடக்கியது. எனவேதான்

shell-ஆனது, DOS-ன் batch files-ஐ விட வலிமை வாய்ந்ததாக கருதப்படுகிறது.

If Statement

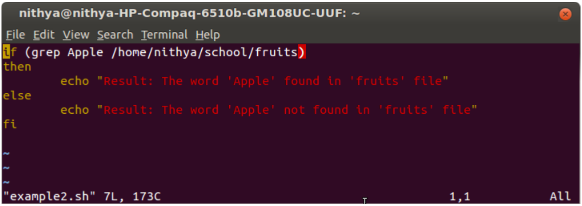
If Construct-ஆனது ஒரு condition-ன் அடிப்படையில் ஒருசில set of commands-ஐ execute செய்ய உதவும்.

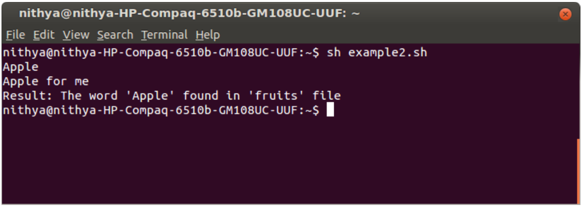
syntax:

```
if (condition)

then

commands

else

commands

fi
```

உதாரணம்:

example2.sh எனும் file-ல் பின்வருமாறு ஒரு if statement-ஐ அடித்துவிட்டு, அதனை run செய்து பார்க்கவும்.





இங்கு fruits எனும் file-க்குள் Apple எனும் வார்த்தை இருப்பின் ஒரு வரியையும் இல்லையெனில் மற்றொரு வரியையும் வெளிப்படுத்துமாறு if command அமைக்கப்பட்டுள்ளது.

மேலும் fruits எனும் file-ஐக் குறிப்பிடும்போது absolute path முறையைக் கையாண்டிருப்பதைக் கவனிக்கவும். shell script-க்குள் எப்பொழுதும் absolute path முறையில் files-ஐக் குறிப்பிடுவதே சிறந்தது.

case Statement

Case statement ஆனது நாம் தேர்ந்தெடுக்கும் option-ன்

அடிப்படையில்,

ஒருசில commands-ஐ execute செய்ய உதவும்.

syntax:

case value in

choice1) commands; ;

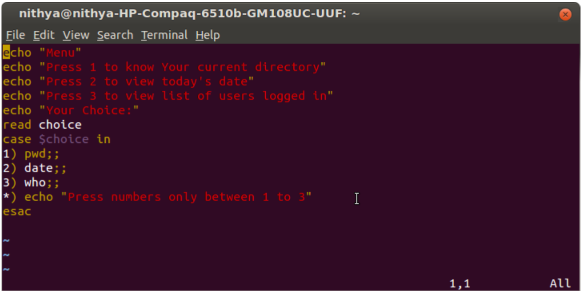
choice2) commands; ;

. . .

esac

உதாரணம்:

example3.sh எனும் file-ல் பின்வருமாறு ஒரு case statement-ஐ அடித்துவிட்டு, அதனை run செய்து பார்க்கவும்.

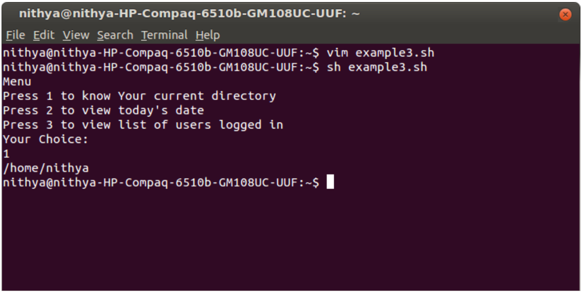



```
echo "Menu"
echo "Press 1 to know Your current directory"
echo "Press 2 to view today's date"
echo "Press 3 to view list of users logged in"
echo "Your Choice:"
read choice
case $choice in
1) pwd;;
2) date;;
3) who;;
*) echo "Press numbers only between 1 to 3"
esac
~
~
~
                                        1,1           All
```

இங்கு முதலில் எந்தெந்த options எதற்கு எனும் விவரம் echo statement மூலம் திரையில் வெளிப்படுத்தப்பட்டுள்ளது. பின்னர் read function மூலம் நம்மிடமிருந்து input பெற்றுக்கொள்ளப்பட்டு, அது choice எனும் variable-ல் சேமித்து வைக்கப்படுகிறது.

பின்னர் **case statement** மூலம் **choice variable**-ல் 1 என இருப்பின் **pwd command**-ஐயும், 2 இருப்பின் **date**-ஐயும், 3 இருப்பின் **who command**-ஐயும் **execute** செய்யுமாறு கொடுக்கப்பட்டுள்ளது. கடைசியாக * என்பது இந்த 1,2,3 எனும் மூன்று எண்கள் தவிர வேறு ஏதாவது இருப்பின், “தயவுசெய்து மூன்று எண்ணுக்குள் ஒரு எண்ணை அழுத்தவும்” எனும் **message** திரையில் வெளிப்படுமாறு ஒரு **echo statement** கொடுக்கப்பட்டுள்ளது.

எனவே நாம் கொடுக்கும் எண்ணை **input**-ஆகக் கொண்டு, இதன் **output** பின்வருமாறு அமையும்.



for Loop

இது ஒரே வகையான செயல்களை ஒரு list-க்குள் உள்ள ஒவ்வொரு மதிப்பின் மீதும் திரும்பத் திரும்பச் செய்து வெவ்வேறு வகையான output-ஐ வெளிப்படுத்தும்.

syntax:

for <variable> in <looping condition>

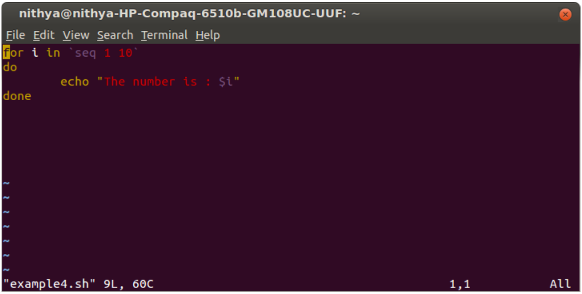
do

Commands

done

உதாரணம்:

example4.sh எனும் file-ல் பின்வருமாரு ஒரு for loop-ஐ அடித்துவிட்டு, அதனை run செய்து பார்க்கவும்.



இங்கு seq-க்கான argument 1 10 என இருப்பதால் இந்த command, 1 முதல் 10 வரையிலான எண்களை வெளிப்படுத்தும்.

இந்த command-ன் இருபுறங்களிலும் backticks பயன்படுத்தப்பட்டு இதன் output-ஆனது for loop-க்கு condition-ஆக கொடுக்கப்பட்டுள்ளது.

எனவே 1 முதல் 10 வரையிலான ஒவ்வொரு எண்ணுக்கும் loop-க்குள் உள்ள statement execute செய்யப்பட்டு, இதன் output பின்வருமாறு அமையும்.

```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~
File Edit View Search Terminal Help
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~$ vim example4.sh
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~$ sh example4.sh
The number is : 1
The number is : 2
The number is : 3
The number is : 4
The number is : 5
The number is : 6
The number is : 7
The number is : 8
The number is : 9
The number is : 10
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~$
```

இதற்கான மற்றுமோர் உதாரணம் பின்வருமாறு.

```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~/scripts
File Edit View Search Terminal Help
for i in `seq 1 10`
do
        echo "Square of $i is `expr $i \* $i`"
done

"example4.sh" 9L, 73C                                  1,1           All
```

```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~/scripts
File Edit View Search Terminal Help

nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~/scripts$ vim example4.sh
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~/scripts$ sh example4.sh
Square of 1 is 1
Square of 2 is 4
Square of 3 is 9
Square of 4 is 16
Square of 5 is 25
Square of 6 is 36
Square of 7 is 49
Square of 8 is 64
Square of 9 is 81
Square of 10 is 100
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~/scripts$
```

while Loop

while loop-ஆனது ஒரு condition சரியாக இருக்கும் பட்சத்தில் loop-க்குள் உள்ள commands-ஐ execute செய்யும்.

syntax:

while <looping condition>

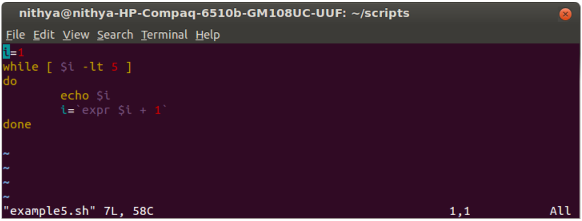
do

Commands

done

உதாரணம்:

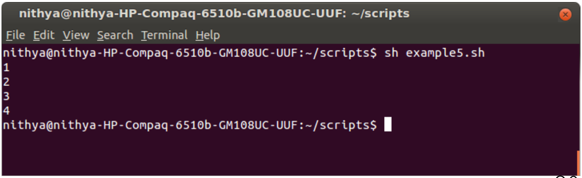
example5.sh எனும் file-ல் பின்வருமாறு ஒரு while loop-ஐ அடித்துவிட்டு, அதனை run செய்து பார்க்கவும்.



இங்கு முதலில் i எனும் variable-க்கு 1 எனும் எண் assign செய்யப்பட்டுள்ளது. பின்னர் while loop-க்குள் கொடுக்கப்பட்டுள்ள condition என்னவெனில், இந்த i எனும் variable, 5 க்கு கீழாக இருக்கும் மட்டும் loop-க்குள்

உள்ளவை **execute** செய்யப்பட வேண்டும் என்பதாகும்.

அடுத்ததாக **loop**-க்குள் இந்த **i** எனும் **variable** ஒவ்வொன்றாக அதிகரிக்கப்படுவதைக் காணலாம். எனவே எப்பொழுது **i**-ன் மதிப்பு 5-க்கு இணையாக வருகிறதோ அப்பொழுது இந்த **loop** முடிக்கப்பட்டு இதன் **output** பின்வருமாரு அமையும்.



until Loop

until loop-ம் while loop-ஐப் போலவே. ஒரே ஒரு வித்தியாசம் என்னவெனில் until loop-ஆனது ஒரு condition தவறாக மாறும் பட்சத்தில் loop-க்குள் உள்ள commands-ஐ execute செய்யாது.

syntax:

until <looping condition>

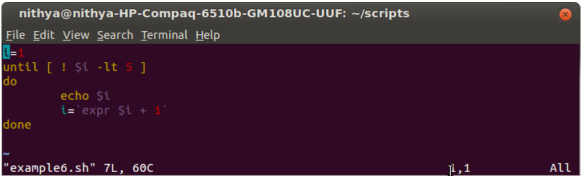
do

Commands

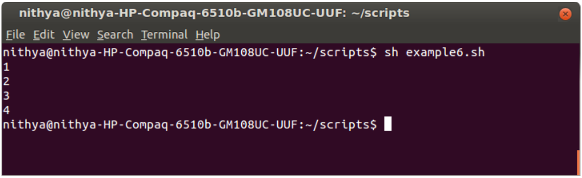
done

உதாரணம்:

example6.sh எனும் file-ல் பின்வருமாரு ஒரு until loop-ஐ அடித்துவிட்டு, அதனை run செய்து பார்க்கவும்.



இதன் output பின்வருமாறு.



Functions

ஒரு shell script-க்குள் உள்ள function என்பது ஒரு சிறு பகுதி

வேலையை செய்து முடிக்க வல்லது. ஒரு பெரிய வேலையை செய்து முடிக்க தொடர்ச்சியாக **commands**-ஐ அடித்துக்கொண்டே செல்லாமல், அவற்றை சிறு சிறு **logical** பகுதிகளாகப் பிரித்து தனித்தனி **function**-ஆக எழுதி வைத்து விடலாம்.

பின்னர் இத்தகைய **functions**-ஐ நமக்குத் தேவைப்படும் இடங்களில் அழைத்து நமது வேலையை சுலபமாக முடித்து விடலாம்.

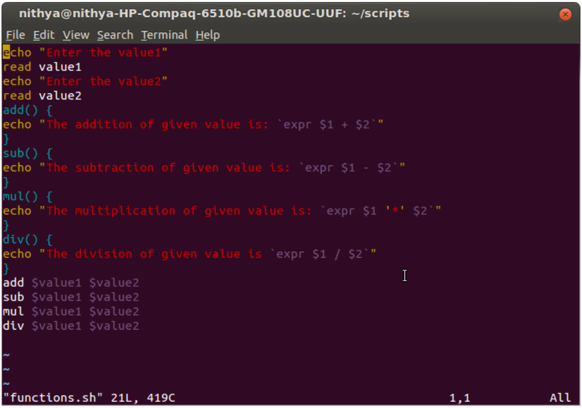
ஒரு சில **function** இயங்குவதற்கு எதேனும் **input** தேவையெனில், அது **parameter** மூலமாக கொடுக்கப்படுகிறது.

Syntax:

```
function_name () {
commands
}
```

உதாரணம்:

functions.sh எனும் file-ல் பின்வருமாரு ஒரு function-ஐ அடித்துவிட்டு, அதனை run செய்து பார்க்கவும்.

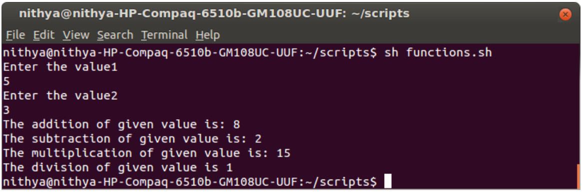


```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~/scripts
File Edit View Search Terminal Help
echo "Enter the value1"
read value1
echo "Enter the value2"
read value2
add() {
echo "The addition of given value is: `expr $1 + $2`"
}
sub() {
echo "The subtraction of given value is: `expr $1 - $2`"
}
mul() {
echo "The multiplication of given value is: `expr $1 '*' $2`"
}
div() {
echo "The division of given value is `expr $1 / $2`"
}
add $value1 $value2
sub $value1 $value2
mul $value1 $value2
div $value1 $value2

~
~
~
"functions.sh" 21L, 419C                                  1,1           All
```

இங்கு முதலில் value1 , value 2 எனும் 2 variables-ஆனது read function மூலம் நம்மிடமிருந்து 2 எண்களை பெற்றுக்கொள்கிறது. பின்னர் அந்த 2 எண்களின் கூட்டலுக்கு add(), கழித்தலுக்கு sub(), பெருக்கலுக்கு mul(), வகுத்தலுக்கு div() எனும் 4 functions வரையறுக்கப்பட்டுள்ளது.

கடைசியாக இந்த functions-ன் இயக்கத்திற்குத் தேவையான 2 எண்களைத் தாங்கியிருக்கும் value1 மற்றும் value2 எனும் 2 variables ஒவ்வொரு function-க்கும் parameter-ஆக கொடுக்கப்பட்டு அவை தூண்டப்படுகின்றன.

இதன் output பின்வருமாறு அமையும்.


nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~/scripts
File Edit View Search Terminal Help
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~/scripts$ sh functions.sh
Enter the value1
5
Enter the value2
3
The addition of given value is: 8
The subtraction of given value is: 2
The multiplication of given value is: 15
The division of given value is 1
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~/scripts$

Job Scheduling

Shell-ல் ஒருசில jobs-ஐ schedule செய்வதற்காக at மற்றும் crontab எனும் 2 commands பயன்படும். இதைப்பற்றி பின்வருமாறு காணலாம்.

at command

ஒரு command/shell script ஒரு நாளில் சரியாக எத்தனை மணியளவில் execute செய்யப்பட வேண்டும் என்பதை முன்கூட்டியே தீர்மானித்து அதனை schedule செய்ய இந்த at command பயன்படுகிறது.

$ touch india | at 16.39

இங்கு உள்ள touch எனும் command சரியாக 4:39 PM-க்கு இயங்கி india எனும் file-ஐ உருவாக்கும்.

crontab

Crontab-ஐப் பயன்படுத்தி ஓர் command/shell

script-ஆனது, குறிப்பாக எந்த நேரத்தில் execute செய்யப்பட வேண்டும் என்பதைத் துல்லியமாகத் தர முடியும்.

crontab-என்பது ஒரு file ஆகும். crontab -e எனக் கொடுப்பதன் மூலம் அந்த file-ஆனது open செய்யப்படும். பின்னர் அதற்குள் கொடுக்கப்படும் விவரங்கள் முறையே பின்வரும் வரிசையில் அமைய வேண்டும்.

- minute – நிமிடம் (0 முதல் 59 வரை)
- hour – மணி (0 முதல் 23 வரை)
- day of month – தேதி (1 முதல் 31 வரை)
- month – மாதம் (1 முதல் 12 வரை)

- **day of week** – கிழமை (0 முதல் 6 வரை) (0=**Sunday**)

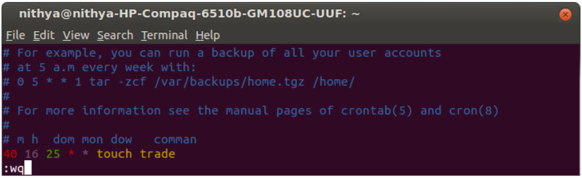
உதாரணத்துக்கு **touch** எனும் **command** ஒவ்வொருமாதமும் 25-ம் தேதி மாலை 4. 40 மணிக்கு இயக்கப்பட வேண்டும் என்று **crontab**-க்குள் கொடுக்க வேண்டுமெனில் அதற்கான **command** பின்வருமாறு அமையும்.

$ **crontab -e**

40 16 25 * * **touch trade**

(press)Esc:wq

(press)Enter



இங்கு முதலில் உள்ள 40 என்பது நிமிடத்தையும், அடுத்து உள்ள 16 மணியையும், பின்னர் உள்ள 25 தேதியையும் குறிக்கிறது. அடுத்ததாக உள்ள ✸ என்பது அனைத்து மாதத்தையும், கடைசியாக உள்ள ✸ என்பது அனைத்துக் கிழமைகளையும் குறிக்கிறது.

crontab-ல் இதுவரை schedule செய்யப்பட்டுள்ள commands அனைத்தையும் பட்டியலிட –l எனும் option-ஐப் பயன்படுத்தவும். இது பின்வருமாறு.

```
$ crontab -l
```

nithya@nithya-HP-Compaq-6510b-GM108UC-UUF: ~

File Edit View Search Terminal Help

```
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~$ crontab -l
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# 0 5 * * 1 tar -zcf /var/backups/home.tgz /home/
#
# For more information see the manual pages of crontab(5) and cron(8)
#
# m h  dom mon dow   comman
40 16 25 * * touch trade
nithya@nithya-HP-Compaq-6510b-GM108UC-UUF:~$
```

Rsync மூலம் backup செய்தல்

நமது கணிணியில் உள்ள கோப்புகளை தூரத்தில் உள்ள வேறு ஒரு கணிணியிலும் சேமித்து வைப்பது நல்லது. நமது கண்ணியில் ஏதேனும் பாதகம் ஏற்பட்டாலும், கோப்புகள் பத்திரமாக இருக்கும்.

rsync எனும் மென்பொருள், இவ்வாறு கோப்புகளை ஒரே கணிணியிலோ அல்லது வேறு ஒரு கணிணியிலோ, எளிய,

சிறந்த முறையில் சேமிக்க உதவுகிறது.

முக்கிய பண்புகள்;

வேகம்:

முதல்முறை இயங்கும் போது sourceல் உள்ள அனைத்து file மற்றும் directory களை destinationக்கு நகல் எடுக்கிறது. அதற்கு அடுத்த இயக்கங்களில், Source ல் மாறுதல் செய்யப்பட்ட file மற்றும் directory களை மட்டுமே நகல் செய்கிறது. இதனால் backup ஆனது, மிக விரைவாக நடைபெறுகிறது.

பாதுகாப்பு:

வேறு ஒரு கணிணிக்கு நகல் செய்யும்போது, ssh என்ற மென்பொருள் வழியாக தகவல் encryption – மறையாக்கம் – செய்யப்பட்டு, பாதுகாக்கப்படுகிறது.

குறைந்த bandwidth:

கோப்புகளை வேறு கணிணிக்கு அனுப்பும் முன், அவை சுருக்கப்படுகின்றன. (compress). அந்த கணிணியில் பெற்றுக்கொண்ட பின், அவை விரிக்கப்படுகின்றன. (extract). இதனால் மிகவும் குறைந்த bandwidth பயன்படுகிறது.

அனுமதிகள்:

எல்லா பயனரும் இதை பயன்படுத்தலாம். root க்கான அனுமதிகள் எதுவும் தேவையில்லை.

Syntax:

rsync <options> <source> <destination>

source மற்றும் destination என்பவை, ஒரே கணிணி அல்லது networkல் உள்ள வேறு ஒரு கணிணியை குறிக்கும். வேறு

கணிணியை குறிக்கும்போது, அதன் பயனர் பெயர், IP முகவரி அல்லது hostname மற்றும் destination full path போன்றவை தரப்பட வேண்டும்.

உதாரணம் – 1;

இரு directory களை ஒரே கணிணியில் sync செய்தல்.

இதற்கு -zvr options களை பயன்படுத்தலாம்.

```
$ rsync -zvr /var/opt/installation/inventory/ /root/t
building file list ... done
sva.xml
svB.xml
.
sent 26385 bytes  received 1098 bytes  54966.00 bytes/sec
total size is 44867  speedup is 1.63
$
```

இதில்,

-z = compress – சுருக்கு

-v = verbose – விரிவான தகவல்

-r = recusrsive – எல்லா கோப்புகள்.

இரு இடங்களிலும் உள்ள கோப்புகளின் timestamp பார்ப்போம்.

```
$ ls -l /var/opt/installation/inventory/sva.xml /root
-r--r--r-- 1 bin bin 949 Jun 18 2009 /var/opt/installati
-r--r--r-- 1 root bin 949 Sep 2 2009 /root/temp/sva.xml
```

இவை ஒன்றாக இல்லை. வெவ்வேறாகவே உள்ளன.

உதாரணம்2: metadata ஐயும் சேர்த்து sync செய்தல்.

Rsync உடன் -a என்ற option ஐ சேர்க்கும் போது பின்வரும் சிறப்பு பண்புகளையும் சேர்த்து sync செய்கிறது.

-a = archive mode

symbolic link

permissions

timestamp

owner, group

மேற்கண்ட உதாரணத்தில் -a சேர்த்து இயக்குவோம்.

```
$ rsync -azv /var/opt/installation/inventory/ /root/t
building file list ... done
```

```
./
sva.xml
svB.xml
.
sent 26499 bytes  received 1104 bytes  55206.00 bytes/sec
total size is 44867  speedup is 1.63
$
```

இப்போது timestamp ம் சேர்த்து sync ஆகி இருப்பதை

காணலாம்.

```
$ ls -l /var/opt/installation/inventory/sva.xml /root
-r--r--r-- 1 root bin  949 Jun 18  2009 /var/opt/installat
```

```
-r--r--r-- 1 root bin 949 Jun 18 2009 /root/temp/sva.xml
```

உதாரணம் 3: ஒரே ஒரு file-ஐ மட்டும் rsync செய்தல்

```
$ rsync -v /var/lib/rpm/Pubkeys /root/temp/
Pubkeys

sent 42 bytes  received 12380 bytes  3549.14 bytes/sec
total size is 12288  speedup is 0.99
```

உதாரணம் 4: local கணிணியில் இருந்து remote கணிணிக்கு sync செய்தல்

remote கணிணியை குறிப்பிடும்போது, பயனர் பெயர், IP முகவரி (அ) hostname, destination directory path ஐ தரவேண்டும். இதன் அமைப்பு username@hostname: /full/path/to/destination.

இப்போது remote கணிணிக்கான password (கடவுச் சொல்) கேட்கப்படும் தந்தபின் rsync நடைபெறும்.

'Passwordless ssh' என்ற அமைப்பை செய்தால், password

கேட்காமலேயே, remote கணிணிக்கு rsync செய்ய முடியும்.

```
$ rsync -avz /root/temp/ shrini@192.168.200.10:/home/shr
Password:
building file list ... done
./
rpm/
rpm/Basenames
rpm/Conflictname

sent 15810261 bytes  received 412 bytes  2432411.23 bytes
total size is 45305958  speedup is 2.87
```

உதாரணம்5: remote கணினியில் இருந்து local கணினிக்கு rsync செய்தல்.

இதற்கு,

source = remote கணினி

destination = local கணினி

என்று தர வேண்டும்.

```
$ rsync -avz shrini@192.168.200.10:/var/lib/rpm /root/te
```

```
Password:
receiving file list ... done
rpm/
rpm/Basenames
.
sent 406 bytes  received 15810230 bytes  2432405.54 bytes
total size is 45305958  speedup is 2.87
```

உதாரணம் 6: remote shell

-e எனும் option மூலம் remote கணிணியில் உள்ல ஒரு shell ஐ நாம் பயன்படுத்தலாம். -e ssh மூலம் பாதுகாப்பான encryption முறையில் rsync நடக்கிறது.

```
$ rsync -avz -e ssh shrini@192.168.200.10:/var/lib/rpm /ro
Password:
receiving file list ... done
rpm/
rpm/Basenames

sent 406 bytes  received 15810230 bytes  2432405.54 bytes
total size is 45305958  speedup is 2.87
```

உதாரணம் 7: Destination-ல் மாறுதல் இருந்தால், overwrite செய்தலை தடுத்தல்.

சில நேரங்களில் destination-ல் உள்ள Fileஐ நாம் மாற்றி இருப்போம். அவை rsync மூலமாக source-ல் இருந்து மாற்றப்படக் கூடாது. இதற்கு -u என்ற option பயன்படுகிறது.

```
$ ls -l /root/temp/Basenames
total 39088
-rwxr-xr-x 1 root root      4096 Sep  2 11:35 Basenames

$ rsync -avzu shrini@192.168.200.10:/var/lib/rpm /root/t
Password:
receiving file list ... done
```

```
rpm/

sent 122 bytes  received 505 bytes  114.00 bytes/sec
total size is 45305958  speedup is 72258.31

$ ls -lrt
total 39088
-rwxr-xr-x 1 root root      4096 Sep  2 11:35 Basenames
```

உதாரணம் 8: வெறும் directory structure-ஐ மட்டும் rsync செய்தல்

வெறும் Directory Tree ஐ மட்டும் (Fileகளை அல்ல) rsync

செய்ய -d பயன்படுகிறது.

```
$ rsync -v -d shrini@192.168.200.10:/var/lib/ .
Password:
receiving file list ... done
logrotate.status
CAM/
YaST2/
acpi/

sent 240 bytes  received 1830 bytes  318.46 bytes/sec
total size is 956  speedup is 0.46
```

உதாரணம் 9: Progress ஐ காட்டுதல்

அதிகமான file-ஐ rsync செய்யும்போது, அதன் நிகழ்வை பார்க்க விரும்புவோம். எத்தனை கோப்புகள் rsync ஆகி உள்ளன. எத்தனை மீதம் உள்ளன என அறிய –progress பயன்படுகிறது.

இதற்கு rsnapshot என்ற மென்பொருளையும் பயன்படுத்தலாம்.

```
$ rsync -avz --progress shrini@192.168.200.10:/var/lib/r
Password:
receiving file list ...
```

```
19 files to consider
./
Basenames
   5357568 100%   14.98MB/s    0:00:00 (xfer#1, to-check=17/19)
Conflictname
     12288 100%   35.09kB/s    0:00:00 (xfer#2, to-check=16/19)
.
.
.
sent 406 bytes  received 15810211 bytes  2108082.27 bytes
total size is 45305958  speedup is 2.87
```

உதாரணம் 10: **Destination**-ல் **delete** செய்தல்

source-ல் ஒரு **file** ஐ **delete** செய்தால், அது ஏற்கனவே **target**-ல் இருந்தாலும் அது **delete** செய்யப்படுவதில்லை. அது அங்கேயே வைக்கப்படுகிறது.

அதையும் **delete** செய்ய **–delete** பயன்படுகிறது.

```
# Source and target are in sync. Now creating new file
$ > new-file.txt
```

```
$ rsync -avz --delete shrini@192.168.200.10:/var/lib/rpm/
Password:
receiving file list ... done
deleting new-file.txt
./

sent 26 bytes  received 390 bytes  48.94 bytes/sec
total size is 45305958  speedup is 108908.55
```

உதாரணம் 11: ஏற்கெனவே உள்ளவற்றை மட்டும் rsync செய்தல்

source ல் புதிய fileகள் இருந்தாலும், அவற்றை விட்டுவிட்டு target ல் ஏற்கெனவே உள்ள fileகளை மட்டும் rsync செய்யலாம். இதற்கு –existing பயன்படுகிறது.

```
[/var/lib/rpm ]$ > new-file.txt
```

```
$ rsync -avz --existing root@192.168.1.2:/var/lib/rpm/ .
root@192.168.1.2's password:
receiving file list ... done
./

sent 26 bytes  received 419 bytes  46.84 bytes/sec
total size is 88551424  speedup is 198991.96
```

உதாரணம் 12: source/destination வித்தியாசத்தை பார்த்தல்

இதற்கு -i பயன்படுகிறது.

Source:

```
$ ls -l /var/lib/rpm
-rw-r--r-- 1 root root 5357568 2010-06-24 08:57 Basenames
-rw-r--r-- 1 root root   12288 2008-05-28 22:03 Conflictnam
-rw-r--r-- 1 root root 1179648 2010-06-24 08:57 Dirnames
```

Destination:

```
$ ls -l /root/temp
-rw-r--r-- 1 root root   12288 May 28  2008 Conflictname
-rw-r--r-- 1 bin bin   1179648 Jun 24 05:27 Dirnames
-rw-r--r-- 1 root root       0 Sep  3 06:39 Basenames
```

```
$ rsync -avzi shrini@192.168.200.10:/var/lib/rpm/ /root/t
Password:
receiving file list ... done
>f.st.... Basenames
.f....og. Dirnames
```

```
sent 48 bytes  received 2182544 bytes  291012.27 bytes/se
total size is 45305958  speedup is 20.76
```

உதாரணம் 13: சேர்த்தலும் நீக்கலும்

ஒருசில கோப்புகளை rsync-ல் சேர்க்கவும், வேறு சிலவற்றை நீக்கவும்

செய்யலாம்.

```
$ rsync -avz --include 'P*' --exclude '*' shrini@192.168.2
```

```
Password:
receiving file list ... done
./
Packages
Providename
Provideversion
Pubkeys

sent 129 bytes  received 10286798 bytes  2285983.78 bytes
total size is 32768000  speedup is 3.19
```

உதாரணம் 14: file size limit

–max-size எனும் option மூலம் இதற்கும் பெரிய file-ஐ rsync செய்வதை தடுக்கலாம்.

```
$ rsync -avz --max-size='100K' shrini@192.168.200.10:/var/l
Password:
receiving file list ... done
./
Conflictname
Group
Installtid
Name
```

```
Sha1header
Sigmd5
Triggername

sent 252 bytes  received 123081 bytes  18974.31 bytes/sec
total size is 45305958  speedup is 367.35
```

உதாரணம் 15: எல்லா file-களையும் sync செய்தல்

source, target ல் வேறுபட்ட file களை மட்டுமே rsync செய்யாமல் எல்லா file களையும் முழுதாக rsync செய்ய -W பயன்படுகிறது.

இதற்கு network bandwidth அதிகம் தேவை. ஆனால் checksum சோதனை தவிர்க்கப்படுவதால், சற்று விரைவாக நடக்கும்.

```
# rsync -avzW shrini@192.168.200.10:/var/lib/rpm/ /root
Password:
receiving file list ... done
./
Basenames
Conflictname
Dirnames
Filemd5s
Group
Installtid
```

Name

sent 406 bytes received 15810211 bytes 2874657.64 bytes

total size is 45305958 speedup is 2.87