Industrial Training Report

"NEK Tech Linux Shell" Development for NEK Tech Labs

NEK Tech
Corporate Office: NEKTech Educational Consulting Pvt. Ltd., 178, M-9, Chitra Complex, MP Nagar Zone I, Bhopal – 462011
Email: service@nektech.in Web: www.nektech.in Contact: +91-77710 43826, +91-755 4224605

TABLE OF CONTENTS

1 INDUSTRY OVERVIEW AND LAB SETUP

1.1 NEK TECH LABS

- *What we are?*

  We are young professionals coming up with the pragmatic idea of carving intellectuals out of engineering academia.

  'NEK Tech' is an establishment of highly intellectual software engineers, who have been working with world's top-notch Software Development companies and are dedicated to provide their knowledge capital to the benefit of the Engineering Institutions. We are experienced in the fields of core Computer Science Product Development, Kernel Programming and Operating Systems Development. The arena, comprising all the mentioned fields, is one of the most lucrative and esteemed, yet the rarest niches to be explored.

- *What we do?*
  - ✓ *Product Development*: "NEK Tech Linux Shell" (Expected to be Dropped in Open Source early 2014)
  - ✓ *Project Development*: "NEK Tech Shell"
  - ✓ *Institution Consulting* for Software R & D: Clients are AIT, Pune; MIT, Pune and Sinhgad Institutions, Pune. Workshop session scheduled for MANIT, Bhopal.

2 TOOLS AND OPERATING SYSTEM USED

2.1 LINUX UNDERSTANDING

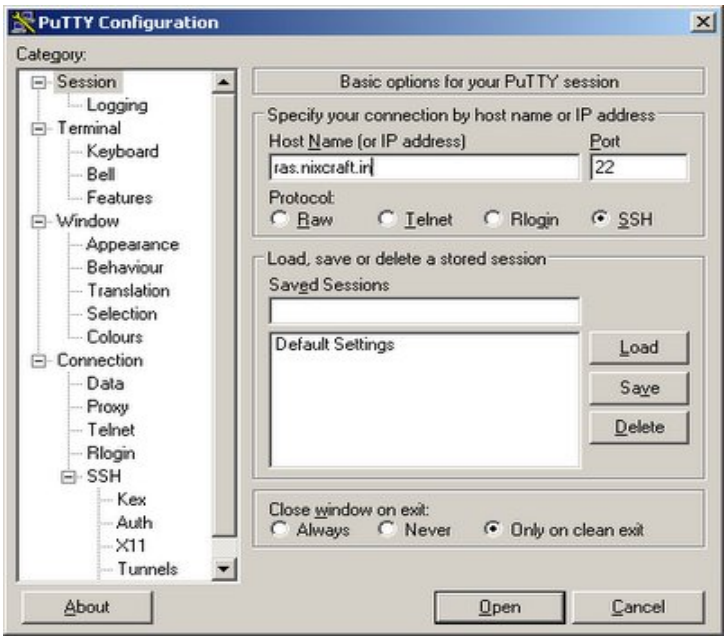
- *Following are the topics Covered in Linux Training*:
  - ✓ Shell Basics
  - ✓ Paths
  - ✓ Navigating the File System
  - ✓ Execution Cycle
  - ✓ Providing Options
  - ✓ Creating Files and Directories
  - ✓ Looking at Files
  - ✓ Basic Tools

Man	Documentation for commands.	mkdir	Make directories.
Cat	Concatenate and display text files.	more	Page through a text file.
Cd	Change working directory.	mv	Move (rename) files and directories.
Clear	Clear the screen.	od	Display the bytes in a file.
Cp	Copy files and directories.	pwd	Print current working directory.
Date	Display the current date and time.	rm	Remove files.
Diff	Show differences between two text files.	rmdir	Remove directories.
echo	Print arguments.	sort	Sort lines.
grep	Print lines matching a pattern.	tail	Display the last few lines of a file.
head	Display the first few lines of a file.	uniq	Remove adjacent duplicate lines.
ls	List files and directories.	wc	Count lines, words, and characters in a file.

  - ✓
  - ✓ Special Characters & Wildcards
  - ✓ Redirecting Input and Output
  - ✓ Redirection Examples
  - ✓ Pipes
  - ✓ File Ownership and Permissions
  - ✓ Directory Permissions
  - ✓ Changing Permissions

2.2 PUTTY SETUP FOR WINDOWS TO CONNECT LINUX SERVER

- What is PUTTY?
  - ✓ You need to use special program called PUTTY. It is a free SSH, Telnet, rlogin, and raw TCP client for Windows system.
  - ✓ Download putty
  - ✓ Download putty.exe from greenend.org.uk. Save putty.exe file on your desktop.
- How do I use putty?
  - ✓ Linux (UNIX) runs special program/service called Secure Shell (SSH) which is designed for logging into system and executing commands on a networked computer. By default ssh server runs on all Linux server. You just need to use putty.exe to connect Linux server from Windows XP.
  - ✓ Double click on putty.exe. A window will open as follows:



- Specify your connection name by typing hostname or IP address (such as 192.168.1.2) of Linux server.
- Click on Open button to connect to remote Linux server via SSH.

- Click on Yes button to accept server's host key and cache the same key
- Type your username (such as pallavi) and password to connect to Linux server

2.3 VIRTUAL BOX SETUP FOR LINUX GUEST OS IN WINDOWS

- Virtual Box is a cross-platform virtualization application.
- What does that mean? For one thing, it installs on your existing Intel or AMD-based computers, whether they are running Windows, Mac, Linux or Solaris operating systems. Secondly, it extends the capabilities of your existing computer so that it can run multiple operating systems (inside multiple virtual machines) at the same time.
- VirtualBox is a wonderful new open source virtualisation product, with a huge range of capabilities, and excellent documentation. As well as having nice enterprise features, for the regular openSUSE user it can be a great tool for trying out new installations or live CDs of openSUSE, and even other operating systems and distributions altogether.
- You can run Windows and Linux on your Mac, run Windows Server 2008 on your Linux server, run Linux on your Windows PC, and so on, all alongside your existing applications. You can install and run as many virtual machines as you like -- the only practical limits are disk space and memory.
- VirtualBox is deceptively simple yet also very powerful. It can run everywhere from small embedded systems or desktop class machines all the way up to datacenter deployments and even Cloud environments.

The techniques and features that VirtualBox provides are useful for several scenarios:

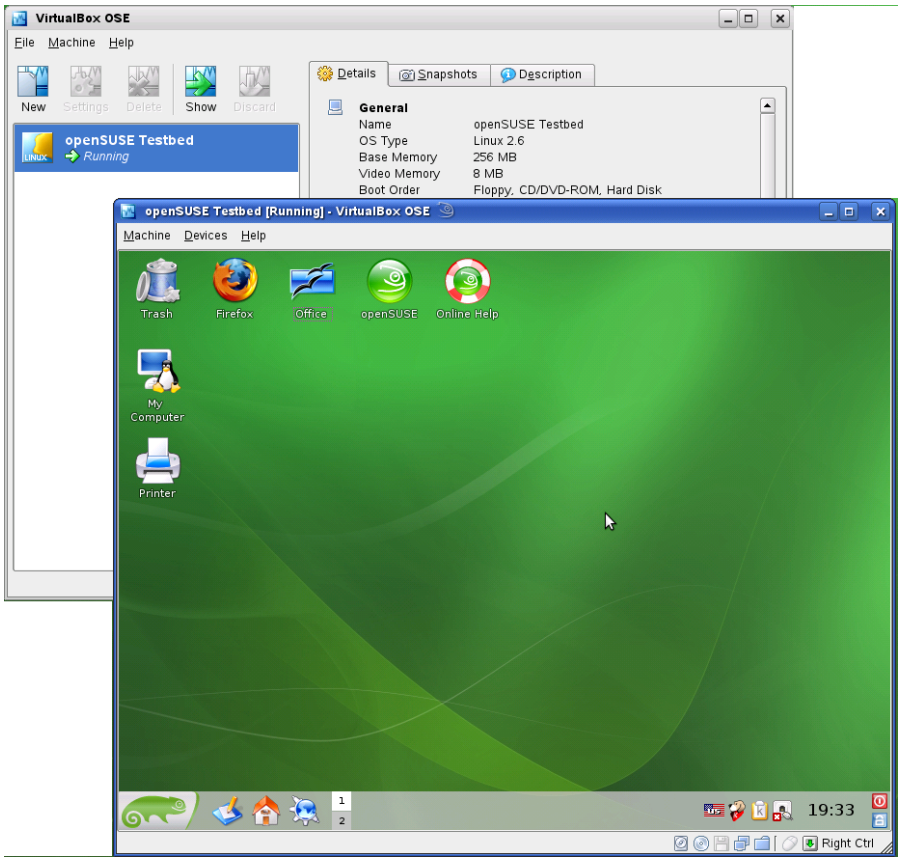
- *Running multiple operating systems simultaneously.* VirtualBox allows you to run more than one operating system at a time. This way, you can run software written for one operating system on another (for example, Windows software on Linux or a Mac) without having to reboot to use it. Since you can configure what kinds of "virtual" hardware should be presented to each such operating system, you can install an old operating system such as DOS or OS/2 even if your real computer's hardware is no longer supported by that operating system.
- *Easier software installations.* Software vendors can use virtual machines to ship entire software configurations. For example, installing a complete mail server solution on a real machine can be a tedious task. With VirtualBox, such a complex setup (then often called an "appliance") can be packed into a virtual machine. Installing and running a mail server becomes as easy as importing such an appliance into VirtualBox.

- *Testing and disaster recovery*. Once installed, a virtual machine and its virtual hard disks can be considered a "container" that can be arbitrarily frozen, woken up, copied, backed up, and transported between hosts.

  On top of that, with the use of another VirtualBox feature called "snapshots", one can save a particular state of a virtual machine and revert back to that state, if necessary. This way, one can freely experiment with a computing environment. If something goes wrong (e.g. after installing misbehaving software or infecting the guest with a virus), one can easily switch back to a previous snapshot and avoid the need of frequent backups and restores.

  Any number of snapshots can be created, allowing you to travel back and forward in virtual machine time. You can delete snapshots while a VM is running to reclaim disk space.

- *Infrastructure consolidation*. Virtualization can significantly reduce hardware and electricity costs. Most of the time, computers today only use a fraction of their potential power and run with low average system loads. A lot of hardware resources as well as electricity is thereby wasted. So, instead of running many such physical computers that are only partially used, one can pack many virtual machines onto a few powerful hosts and balance the loads between them.

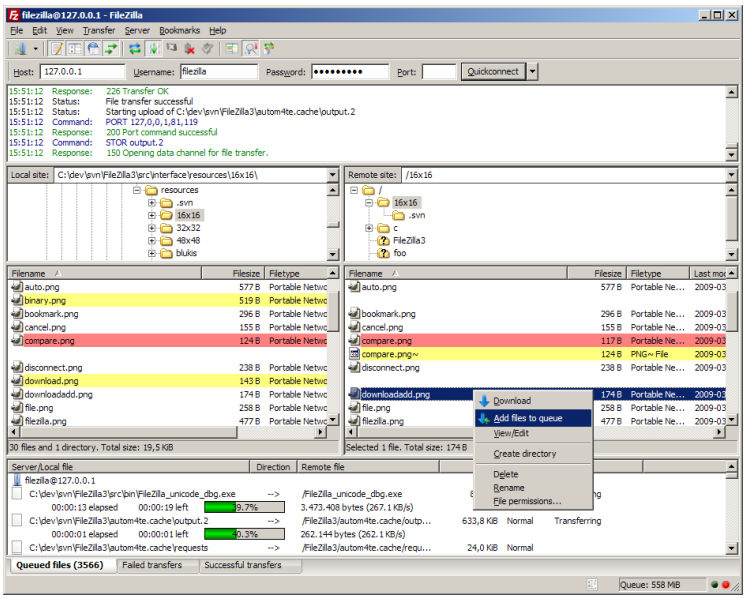

VirtualBox OSE
File Machine Help
New Settings Delete Show Discard
openSUSE Testbed
Running
Details Snapshots Description
General
Name openSUSE Testbed
OS Type Linux 2.6
Base Memory 256 MB
Video Memory 8 MB
Boot Order Floppy, CD/DVD-ROM, Hard Disk
openSUSE Testbed [Running] - VirtualBox OSE
Machine Devices Help
Trash
Firefox
Office
openSUSE
Online Help
My Computer
Printer
19:33
Right Ctrl

2.4 FILE ZILLA SETUP (SFTP TOOL FOR FILE TRANSFER)

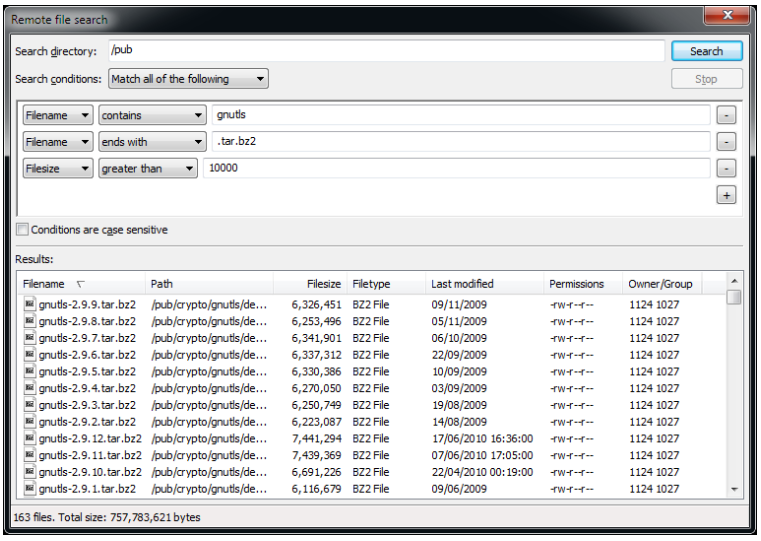
- FileZilla Client is a fast and reliable cross-platform FTP, FTPS and SFTP client with lots of useful features and an intuitive graphical user interface.
- Among others, the features of FileZilla include the following:

  - ✓ Easy to use
  - ✓ Supports FTP, FTP over SSL/TLS (FTPS) and SSH File Transfer Protocol (SFTP)
  - ✓ Cross-platform. Runs on Windows, Linux, *BSD, Mac OS X and more
  - ✓ IPv6 support
  - ✓ Available in many languages
  - ✓ Supports resume and transfer of large files >4GB
  - ✓ Tabbed user interface
  - ✓ Powerful Site Manager and transfer queue
  - ✓ Bookmarks
  - ✓ Drag & drop support
  - ✓ Configurable transfer speed limits
  - ✓ Filename filters
  - ✓ Directory comparison
  - ✓ Network configuration wizard
  - ✓ Remote file editing
  - ✓ Keep-alive
  - ✓ HTTP/1.1, SOCKS5 and FTP-Proxy support
  - ✓ Logging to file
  - ✓ Synchronized directory browsing
  - ✓ Remote file search

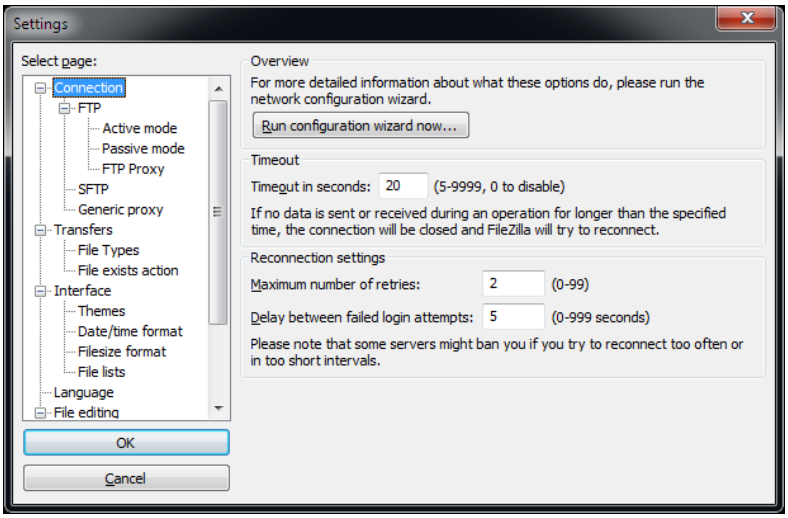
- FileZilla Screenshots:
  - ✓ Login and Front Window



- ✓ Remote File Search



- ✓ Settings and Preferences



- FileZilla Installation for Windows
  - ✓ These instructions constitute the recommended procedure for installing and configuring the FileZilla FTP client. Start by downloading the latest version of the FileZilla FTP client from FileZilla-Project.org. Remember the location to which you download the file.
  - ✓ Once the executable file is downloaded, run it to commence the installation routine.
  - ✓ During the installation process, select the default (Standard) install and retain all other default settings as you continue installation.
  - ✓ Once FileZilla is installed, proceed to the configuration.

2.5 INSTALATION OF LINUX SYSTEM:

- Steps for installation of Linux Systems:
  - ✓ Download Fedora/Ubuntu/Other Linux from Internet according to the architecture of your systems. It could be X86/X86_64/AMD64. (Execute wget http://download.fedoraproject.org/pub/fedora/linux/releases/19/Live/x86_64/Fedora-Live-Desktop-x86_64-19-1.iso)
  - ✓ Download Live USB Creator. (https://fedorahosted.org/liveusb-creator/)
  - ✓ Install Live USB Creator on windows.
  - ✓ Insert a removable Pen Drive.
  - ✓ Create a bootable USB from the ISO download from fedora website.
- Reboot the system and change the bios setting so that it will start boot from USB Drive.
- Once it is booted from USB stick. Please start the Installation application. Select drive and storage options and start the installation.

2.6 VI EDITOR

- *vi/vim* is the most commonly available screen editor for UNIX. It's the only one you can count on being installed on almost every Linux/Unix system.
- *Editing Modes*: The most important thing to know about vi (and the most confusing) is that it has two modes, Command Mode and Insert Mode.
- In Command Mode, you can invoke editing commands, move the cursor, save or exit the file, invoke the shell, or enter Insert Mode.
- In Insert Mode, you can insert new text.
- By default, vi starts in Command Mode.
- *vi* is universally available on Unix systems. It has been around so long in a stable form that it is essentially bug free. Many clones have been written for other kinds of computers.
- *vi* has many powerful commands that utilize just the alphanumeric keys -- it does not require special function keys.
- *vi* is a small program that does not require a lot of system memory or CPU time. It works very fast, even on large files.
- While *vi* is not programmable, it has a simple way to let other Unix programs, such as the sort utility, work on selected portions of your file. This adds the functionality of all those programs to the editor.
- *vi* is completely terminal device independent. It will work with **any** kind of terminal.A system file describes the capabilities and control sequences of each kind of terminal for *vi*. All the program needs to know is what type of terminal you have. When you log in, if pangea cannot figure out what kind of terminal you have, it will prompt you to specify a terminal type. The most common type is the vt100, which most modern terminals and PC communications software emulate.
- The chief disadvantage of *vi* is that it is touchy. That is, every single key you touch on the keyboard seems to do something, often something mysteous. There is a rich set of single character commands to learn.

2.7 GCC COMPILER

- The *GNU Compiler Collection* (GCC) is a compiler system produced by the GNU Project supporting various programming languages.
- Originally named the *GNU C Compiler,* because it only handled the C-Programming language.
- GCC's external interface is generally standard for a UNIX compiler. Users invoke a driver program named gcc, which interprets command arguments, decides which language compilers to use for each input file, runs the assembler on their output, and then possibly runs the linker to produce a complete executable binary.
- gcc is a separate program that reads source code and outputs machine code.
- A per-language front end parses the source code in that language and produces an abstract syntax tree ("tree" for short).
- Example commands:
  - ✓ *gcc –o <output-file> <c-source_file.c>*
  - ✓ *gcc <c-source_file.c>*

2.8 THE MAKE TOOL

- Make is a tool, which controls the generation of executable and other non-source files of a program from the program's source files.
- Make gets its knowledge of how to build your program from a file called the *makefile*, which lists each of the non-source files and how to compute it from other files.
- When you write a program, you should write a "Makefile" for it, so that it is possible to use Make to build and install the program.
- Make enables the end user to build and install your package without knowing the details of how that is done -- because these details are recorded in the makefile that you supply.
- Here is what a simple rule looks like:

```
target:   dependencies ...
           commands
           ...
```

- For example "NEK Tech Shell" Compilation Makefile is:

```
nektech@nektech-desktop:~/prog/nektech_shell$ cat Makefile
all: nektech_shell

nektech_shell: input_parser.o cmd_execution.o
   gcc input_parser.o cmd_execution.o -o nektech_shell

cmd_execution.o: cmd_execution.c
   gcc -c -g cmd_execution.c

input_parser.o: input_parser.c
   gcc -c -g input_parser.c

clean:
   rm -rf *.o nektech_shell
install:
   cp ./nektech_shell /bin
```

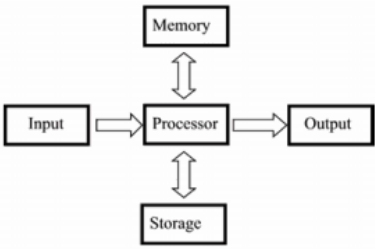
2.9 CSCOPE CODE BROWSER

- cscope is a Linux tool for browsing source code in a terminal environment. C was originally built to work with C code, but also works well with C++, Java, and some other languages.
- Install cscope on Linux system by firing following command:
  - ✓ For debian based systems: *$sudo apt-get install cscope*
  - ✓ For Red Hat based systems: *$sudo yum install cscope*
- Change directory to Source code directory and run "cscope –R". It will open a tool for c-source browsing:

2.10 SYSTEM HARDWARE

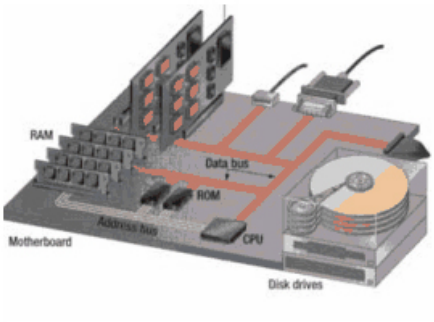
- A computer is made up of Hardware.
- Hardware is the physical component of the computer system. E.g. Monitor, Key Board, mouse, CD/DVD ROM, RAM, CPU etc.
- High Level View of Computer System:

High Level View Of A Computer

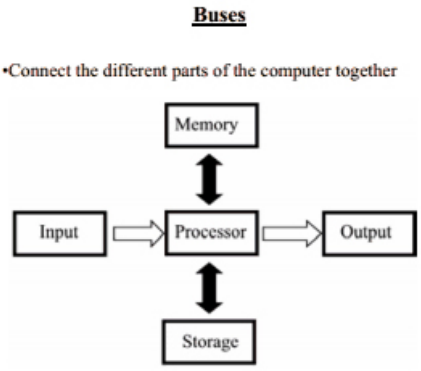


- *Buses* connects the different part of Computer altogether:

Buses (2)



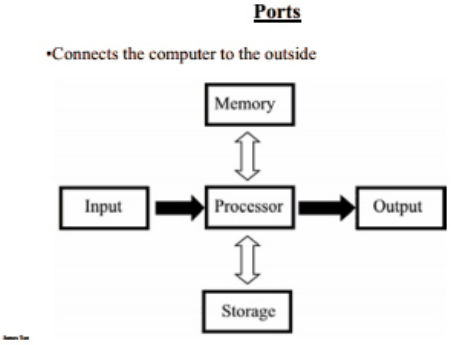
Buses



- *Ports* connects the the Computer to outside:

Ports

Ports



- *Memory* (e.g. RAM)
  - ✓ Keeps the information for the shorter period of time (Usually Volatile).
  - ✓ Faster
  - ✓ More expensive

- *Storage* (e.g. Hard Drives)
  - ✓ The information remains for longer period of time (Non-Volatile).
  - ✓ Slower
  - ✓ Cheaper

2.11 REFERENCES:

http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html
http://rcc.its.psu.edu/user_guides/remote_connectivity/putty/
https://www.virtualbox.org/manual/ch01.html
https://filezilla-project.org/client_features.php

3 LIVE PROJECT ON "LINUX SHELL DEVELOPMENT"