# Linux Platforms as a Secure Desktop Solution

Mohammed A. Afifi, PhD
School of Engineering &
Information Technology
Al Dar University College,
Dubai, United Arab Emirates

Khawar Nehal
Applied Technology Research
Center (ATRC), Karachi, Sindh,
Pakistan

## ABSTRACT

A lot of discussion goes into Enterprise Security, Network Security, and Computer Security. Initially, the most commonly used software are a combination of Operating systems, Desktop, and several standard applications like email clients, web browsers, word processing for document creating and editing. Now, the fact is that the listed common software combinations are the target of the most social engineered attacks, so selecting a secure combination for desktops shall go way far from these attacks into raising the security levels in many organizations. In this paper, we shall look at some of the configurations available in Linux platform that allows it to avoid and mitigate many of these common attacks which are considered to be the main cause of many security breaches.

## General Terms

Linux platform, Desktop Solutions, Desktop Operating Systems, Desktop Security, Computer Security, PC Security, Linux Security.

## Keywords

Linux, Linux platform, Linux Distributions, PC Linux Operating System (PCLOS), Desktop Solution, Secure Desktop, Operating Systems, Network Security, Computer Security, UNIX, PC Security, Android, Windows, MacOS.

## 1. INTRODUCTION

Bell Labs is biased towards open systems and publicly available standards. As the inventors of UNIX, we have a large and stable UNIX environment. When PCs first appeared they were often castigated to separate networks for security reasons and we required users to support themselves. Demand for official support grew at the same time as self-administered machines were causing havoc. Finally, we decided to give them official support to limit their damage. [1]

If end users are to select among different desktop platforms, several desktops are available, including but not limited to Android, Linux, Windows, MacOS, and others. We shall discuss the security issues related to desktop security. Also suggestions shall be made based on experience related to reducing the effects of malware while using the least amount of administration and configuration resources.

The evaluations are of various types that include:

- Out of the box (Ubuntu performance.)
- Hardened in reality (SE Linux)
- Legal loopholes analysis (example EULA)
- Theoretical analysis of security based on documentation. (example EAL4)

End users want a desktop that is user-friendly (easy to use), having a variety of applications that serves their needs, and secure. We cannot force the end users to use a particular desktop but draw their attention to the recommended one with incentives of security and reliability.

## 2. HOW TO SECURE DESKTOPS?

There are multiple steps that can be taken to reduce malware and its effects.

### 2.1 Network Security

Network security is a very broad term where it has a number of security implications; here are the most important security issues settings that will help to extremely improve the network security.

### *2.1.1 Router Settings*

Network Address Translation (NAT); This allows the desktops to access IP addresses and ports outside. However, the outside machines cannot directly access any port or IP address inside the LAN.

### *2.1.2 Proxy Server and Transparent Proxy*

All web pages and DNS requests need to be transparently routed through the transparent proxy server and the local DNS server. This prevents any packets trying to act like TCP port 80 http connections to go out. The proxy server also should deny all requests to black-listed-sites and IP addresses. The black-listed IP address needs to be taken from SPAM Block Lists and Phishtank, where Phishtank and other phishing sites can be integrated into the squid proxy server with the software ufdbguard.

DNS requests can be transparently redirected and the named service on the local server can be set to query OpenDNS servers. This way OpenDNS shall reduce the chances that queries to phishing and malware domains are resolved into IP addresses. Software needs to be added which lists all IP addresses related to malware domains into the squid proxy block list. This is based on the fact that the phishing sites could be based on multiple domains on the same IP.

### 2.2 Software Security via Executable File Integrity

The antivirus software which is based on submissions from the users can be used to detect viruses. The submissions come from users who could be using any brand of antivirus. If any new virus is found, then it is submitted to the antivirus software whose database accepts submissions. This antivirus can be set to send email reports to the administrators. Software needs to be monitoring the reports from all the desktops to determine weak desktops which have a higher chance of being infected by a detectable virus. This indicates a weak desktop.

Many viruses cannot be detected by any antivirus and the number of viruses continues to increase into tens of thousands. The need is to make sure the executable files are not modified. Checksums are to be used and any (.exe) file that is modified needs to be automatically replaced with the original. The modified file can be copied into a suspected infection directory for future analysis. There is a tool named The File Checksum Integrity Verifier which can be automated to check all files in a windows based machine.

"The File Checksum Integrity Verifier (FCIV) is a command-prompt utility that computes and verifies cryptographic hash values of files. FCIV can compute MD5 or SHA-1 cryptographic hash values. These values can be displayed on the screen or saved in an XML file database for later use and verification." [5]

For Linux there is md5sum. If the files are checked for any modifications, then the chances of a virus entering and executing without making changes to executable files becomes very remote and rare.

Another way to protect applications is to create a separate directory in the file servers for each application to be installed. Then install the application to the separate directory and instruct the file server to make the directory read only. Since the directory permissions are now in the control of another machine, it is not a trivial thing to enter a desktop and infect the executable application because the directory is read only.

The file server can be hosted in a SaMBa based system. This shall ensure that if a virus can infect a Windows machine, it would have to be a very capable multiplatform virus to be able to infect a Linux based server to infect any executable files.

### 2.3 Single Act Execute

Linux and UNIX security is well demonstrated and tested based on the fact that telecom and internet companies use it to run their infrastructure successfully on it. No execution will take place unless the administrator deliberately allows it. Continuing to rely on a secure platform for as many server and desktop related services helps reduce the chances of successful penetrations into the systems used.

## 3. THE POWER OF LINUX-BASED DESKTOPS

There are several reasons why Linux machines have way lower infections, almost nothing serious hence secured, than other common platforms. The followings state some of the important features of Linux desktop for an end user:

1. The configuration settings do not allow easy access to the super user (root account), unless deliberately has been used.

2. The super user is not accessible without the user's consent to any software. This is maintained by the experience of being a multiuser system than being used as a single user system. A multiuser system has very well tested and secured super user facilities.

3. The system shall check on the source and it has to be known else the user will be warned that this file is of an unknown source.

4. Instead of having a number of piled applications over each other, multiple desktops can be used to group these applications where it will be much easier and faster to switch between them.

5. The stability adds another important value to the desktop where you barely (I have never experienced any) see the famous application crash popup asking you if you want to send a report.

6. Most of the commonly used desktops are Linux-based as of writing this paper (check the one you have in your smart phone), it is one of the fastest growing and still counting.

7. More over malware will never be able to harm your desktop as it cannot do anything to the system files in particular and any other file in general without the system password. So by simply not providing the password once suspicious problems will be simply avoided.

8. Loopholes to super user access are identified and closed strictly and are considered a serious issue. This is the reason for the success of multiuser systems in the telecom and ISP environment.

9. Linux desktops do not need an antivirus software, as explained earlier that there is no single act execution without the deliberate permission using the super user credentials which makes Linux desktops well-protected and secured.

There has not been (I have never experienced or heard of, did you?) a single famous Linux infection of the types common on other desktops and operating systems. On the other hand, Linux desktop is not immune against security breaches but for many reasons, its system attributes will keep it safe and sound for years to come.

## 4. CONCLUSION

Most of the other desktops were designed first to be used as a single user desktop machine. If the single user was able to access super user functions, it was never considered too much of an issue because the functionality and ease of use was the top priority.

Restricting access to functions was never and still is not a top priority in a single user machine. After all, the super user (root) and the user are usually the same user. The attitude to security design is not apparent to many IT administrators who get to see mainly the heavy marketing advertisements which continuously state that a single user design is secure. The fact is that designing a single OS software to be used as a single user desktop and also as a server shall split the designers towards lax security and make them prefer ease of access to the administrator functions. In this process many loopholes which allow the malware to access the super user functions without needing approval go unnoticed.

If a software is designed to first restrict all super user access to normal users, then it shall be a real multiuser machine. However, this design shall prevent ease of use to the ordinary user. This is evident when Linux machines clearly have all applications stating that they need to have the user talk to the system administrator to modify anything which relates to super user access functions. This includes simple tasks like changing the date and time of the desktop clock. This is because the changing of the desktop clock calls the changing of the system clock.

We hope this example illustrates why technical people understand and consider Linux and UNIX systems as more reliable and secure as compared to single user design systems like many others. They know and can see the ease of disastrous infections in the single user design almost on a daily basis.

## 5. REFERENCES

[1] Thomas A Limoncelli, Robert Fulmer, Thomas Reingold, Alex Levine, and Ralph Loura. Aug 1998. Providing Reliable NT Desktop Services by Avoiding NT Server: Lucent Technologies, Bell Labs. https://www.usenix.org/legacy/publications/library/proceedings/lisa-nt98/full_papers/limoncelli/limoncelli.pdf

[2] UK Government. April 2014. Government Security Classifications.https://www.gov.uk/government/uploads/system/uploads/attachment_data/file/251480/Government-Security-Classifications-April-2014.pdf

[3] UK Government. 2014. UK Gov. Security Assessment puts Ubuntu in First Place. https://insights.ubuntu.com/wp-content/uploads/226b/UK-Gov-Report-Summary.pdf

[4] Microsoft Corporation. June 2013. The File Checksum IntegrityVerifier.http://support.microsoft.com/kb/841290