OPEN ACCESS

International
Journal
Of Modern Engineering Research (IJMER)

Design and Configuration of App Supportive Indirect Internet Access using a Transparent Proxy Server

Pranjal Sharma[1], T. Benith[2]

[1]*Electronics and Communications Engineering, National Institute of Technology, Bhopal, India,*
[2]*Electronics Engineering, Indian Institute of Technology (BHU), Varanasi, India*

***Abstract**: Nowadays apps satisfy a wide array of requirements but are particularly very useful for educational institutions trying to realize their mobile learning systems or for companies wishing to bolster their businesses. A company/institute that wants to perform web filtering, caching, user monitoring etc. and allow Internet access only after authentication might use an explicit proxy. It has been observed that most of the apps that need to connect to the Internet through an explicit proxy, do not work whatsoever. In this paper, a solution has been proposed to get the apps working without having to avoid the use of a proxy server. The solution is developed around transparent proxy and makes use of a captive portal for authentication. Oracle VM VirtualBox was used to develop a test bed for the experiment and pfSense was used as the firewall which has both proxy server and captive portal services integrated on a single platform. When tested, Windows 8 apps as well as Ubuntu apps worked well without sacrificing proxy server services such as web filtering. The proposed solution is widely applicable and cost-effective as it uses open source software and essentially the same hardware as used for explicit proxy deployments.*

***Keywords:** App, Captive portal, Firewall, Proxy server, Transparent proxy.*

I. Introduction

Since their dawn, apps have seen an exponential rise in their use and have almost become ubiquitous. There are millions of apps that satisfy a wide range of requirements, including the most trivial ones. But in some environments, apps can play a key part in the achievement of bigger goals. If an organization also wants to use proxy server services such as caching, web filtering, user monitoring etc. and allow only authorized access to the Internet, it might set up an explicit proxy (also called a direct proxy) [1] as an obvious choice. The choice is so obvious as it combines the previously mentioned proxy services with authentication, in one scheme. But one of the problems with explicit proxies is that most of the apps fail to work because, such apps are usually designed under the assumption that there is an uninterrupted path out to the Internet. This may happen either because the app does not use the explicit proxy setting (configured at the client end) or because the app has no provision to be able to use an explicit proxy to connect to the Internet. [1]

Generally, to get the apps to work properly, an organization may avoid using a proxy server itself, which in turn renders it unable to perform any web filtering, user monitoring or even caching unless it is willing to pay for the more expensive network security solutions such as UTMs (Unified Threat Management Systems) or NGFWs (Next Generation Firewalls). As an example of the problem, the native apps in Windows 8 do not work when using an explicit proxy to connect to the Internet. Microsoft has resolved this issue in Windows 8.1 and even with explicit proxies, the apps work just fine. But in environments, where machines may need to run different OSs (let alone different versions of an OS), a solution is needed which gets all the apps to work (without having to avoid the use of a proxy server itself), regardless of their platform.

In this paper, a solution is proposed which gets the apps to work seamlessly without losing any proxy server functionalities. The solution makes use of a transparent proxy, primarily to get the apps to work. Since a transparent proxy is being used, a user cannot be challenged for credentials by the proxy server itself, as the web browser is not aware of proxy server's existence. Therefore a captive portal is being used for user authorization. To verify the expected results, a test bed was developed using VirtualBox [2] in which client machines connect to the Internet through a pfSense firewall [3] which includes both proxy server and captive portal functionalities. After setting up a Squid [4] transparent proxy and a captive portal in the given test bed, Windows 8 native apps as well as Ubuntu 13.10 apps were tested and they worked flawlessly.

The rest of the paper is organized as follows: Section 2 describes the key components of the proposed solution, Section 3 deals with the development of a test platform using VirtualBox, Section 4 includes details about the implementation of the solution, Section 5 has details about the test performed and the results obtained, Section 6 lists the possible scenarios for the deployment of the solution. Finally, Section 7 concludes.

II. Key Components Of The Solution

The proposed solution involves using a transparent proxy to fulfill the filtering, caching and monitoring requirements. The solution also uses a captive portal to serve the purpose of user authentication, since the use of transparent proxy makes it impossible to authenticate using the proxy itself (Users are not aware of the proxy's existence and hence can't be challenged by the proxy for their credentials). Therefore, the two key topics, 'Transparent Proxy' and 'Captive Portal' are discussed in the following subsections.

2.1 Transparent Proxy

According to RFC 2616 [5], a proxy server is an intermediate program which acts as a server as well as a client to make requests on behalf of the actual clients. It allows client computers to make indirect network connections to other network services. Clients connect to the proxy server and request some resources like web pages, videos etc. On getting the request, the proxy server will check the cache in its local hard disk, for the resources. If the resources have been previously cached, the proxy server will return them to the clients, else it will connect to the relevant remote servers and request the resources on behalf of the clients. It will then cache the resources returned by the remote servers, to serve any subsequent requests for the same resources locally from its cache. There are two main types of proxies [6] in use by client computers, explicit proxy (or direct proxy) and transparent proxy. For the explicit proxy, individual client browsers have to be configured (either manually or using a PAC file) to send requests directly to the proxy server. The disadvantages inherent to this approach include:

- The ability of a user to bypass the proxy by simply altering the client proxy configuration.
- The absence of a direct path for software applications out to the Internet, thereby not allowing them to work properly, as previously discussed in Section 1.

In the transparent proxy deployments, the user's client software (typically a web browser) is unaware that it is communicating with a proxy. A Transparent proxy does not require any configuration on the client's end and usually makes use of efficient forwarding mechanisms such as GRE tunneling, NAT, Cisco's WCCP protocol, or MAC rewrites to direct users to them, automatically. [7] Clients request Internet resources as usual and the transparent proxy serves their requests. The proxy establishes a connection with the desired server and returns requested content to the client as if it came directly from the origin server. A transparent proxy is generally placed in-line between the client and the Internet.

Transparent proxy servers find themselves as ideal choices for web accelerators and web filtering gateways, since client machines are not aware of their presence. [7] In fact, Most ISPs prefer transparent caching proxies as these caches require no configuration at the client end. [8] Another advantage is that, all the software applications work seamlessly, as there is an uninterrupted path out to the Internet. [1]

The following diagram shows a likely configuration for a company that wishes to monitor its employees, employ caching or/and perform web filtering using a transparent proxy.

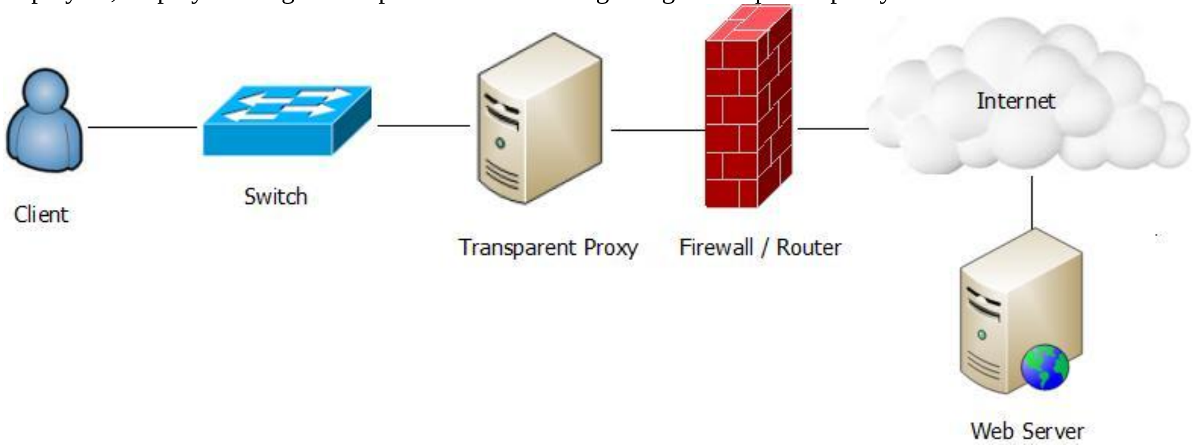


Fig 1: A Possible configuration for transparent proxy deployment

2.2 Captive Portal

The captive portal technique attempts to prevent users from accessing network resources (usually Internet access) until they have authenticated with a server (called as Authentication Server). It is a mechanism that allows a web browser to be used as an authentication device. A user that wishes to access the network, opens up the web browser and tries to access the web. The user is then redirected to a web page that may either present use policy or challenge the user for valid credentials. After successful login or acceptance of use policy, the user is allowed to use the network normally. All the unauthorized users, however, are redirected to the captive portal page, transparently. [9]

Captive portals make use of dynamic firewalling. By default, all access is denied. When a user tries to connect to a server, he/she must be authenticated and is thus redirected to the authentication server. The connection to the authentication server must be secure so as to protect confidential information such as passwords. If the client authenticates successfully, the authentication server notifies the firewall of the same and the firewall rules change dynamically to grant Internet access to the user. Here, the authorization server acts as a central repository for valid user credentials. [10] In a nutshell, to establish a captive portal authentication mechanism, we need:

- A firewall
- A redirection mechanism for web based traffic
- A secure mechanism for user login
- A database for users' credentials

Since authentication in schemes utilizing a captive portal is web based, it becomes necessary to have web browsers on our devices that we wish to use, to access the Internet. This is one of the few drawbacks of an otherwise advantageous authentication mechanism.

III. Development Of A Test Bed Using VirtualBox

We have used VirtualBox to create a logical network setup, in which, the clients connect to the Internet through a pfSense firewall.

3.1 VirtualBox

Oracle VM VirtualBox is a cross-platform hypervisor (virtualization application). It can extend the capabilities of an existing computer system and allow it to run multiple operating systems (inside multiple virtual machines) simultaneously. So, for example, we can run Mac and Linux on our Windows, run Solaris and BSD on Linux and so on, alongside the host machine's existing applications. The number of virtual machines that can be installed and run is limited only by disk space and memory. [11]

3.1.1 Virtualization

Virtualization refers to the act of creating a virtual version of something, including a virtual computer hardware platform, operating system, computer network resources or storage device. A virtual computer is a logical computer (existing in software) with almost all the capabilities of a physical computer. [12] Hypervisors such as VirtualBox, hide physical machine's resources so that they can be shared among multiple virtual machines.

3.1.2 Terminologies related to VirtualBox

- **Host operating system (Host OS):** This is the operating system of the physical computer on which VirtualBox is installed. VirtualBox is available for Windows, Linux, Mac OS X and Solaris host operating systems.

- **Guest operating system (Guest OS):** This is the operating system running inside the virtual machine. Suppositionally, any x86 operating system (DOS, Windows, OS/2, FreeBSD, OpenBSD) can be run on VirtualBox but certain operating systems are optimized to perform better. The select few, however, include the most common ones.

- **Virtual machine (VM):** It is a special environment created by VirtualBox for the guest OS while it is running. Thus, a guest OS runs in a VM. VirtualBox considers a VM as a collection of parameters that characterize its behavior. These parameters include hardware settings and state information about the VM.

3.1.3 Networking in VirtualBox

For each VM, VirtualBox provides up to eight virtual PCI Ethernet cards. Each of the eight networking adapters can be separately configured to operate in one of the following modes:

- **Not attached:** In this mode, the guest machine behaves as if a network card is present, but there is no network connection.

- **Network Address Translation (NAT):** When the NAT mode is enabled for a VM, it acts like a normal computer that connects to the Internet via a router. Here, the VirtualBox networking engine acts as a router. When the guest OS boots, it particularly uses DHCP to obtain an IP address automatically. VirtualBox will tell the VM its assigned IP address. In this mode, every guest is assigned the same IP address, as each of the VM thinks that it is on its own private (isolated) network. The VirtualBox networking engine rewrites every packet from the VMs to appear as though they originated from the host machine, rather than the guest machine. The disadvantage of this mode is that, the VM is unreachable from the outside internet, much like a device in a private network. Therefore, a server cannot be run on a VM unless port forwarding is being used. [11]

- **NAT Network:** Network Address Translation (NAT) Network mode allows VMs to talk to each other on the same host, and communicate with the outside world.

- **Bridged networking:** This mode should be used when the VM needs to be treated as equal to the host on a network i.e. the VM can access all the network services that the host can, for e.g. external DHCP services. In this mode, a virtual NIC is bridged to a physical NIC on the host. Therefore, the VM will get connected to the network that the host machine is connected to. [13]

- **Internal networking:** The internal network is a completely isolated network and not even the host is a member of this network. In this mode, VirtualBox doesn't provide DHCP services and thus, the machines must be statically configured. If required, we can even configure VMs to have multiple NICs that have internal and other network modes thereby providing routes.

- **Host-only networking:** All the VMs residing on the 'Host-only network' can reach each other. In addition, the host can reach the VMs too. However, other external machines cannot reach the VMs on the 'Host-only network', hence the name "Host-only". As the host is now on the same network as the guests it can provide DHCP services. This mode is like a hybrid of 'Bridged networking' and 'Internal networking' modes. [11]

- **Generic networking:** This mode includes sub-modes to either connect guests running on different hosts or to connect to a VDE (Virtual Distributed Ethernet) switch on a Linux or a FreeBSD host.

3.2 Test Bed Setup

We have used VirtualBox to create the logical (virtual) network setup. For this setup, three Guest OSs (pfSense 2.1.X, Ubuntu 13.10, Windows 8) were installed in VirtualBox. For pfSense 2.1.X VM, two network adapters were enabled, one set to 'NAT' and the other to 'Internal Network'. However, for Ubuntu 13.10 VM and Windows 8 VM, only one network adapter was enabled and set to 'Internal Network'.

The VirtualBox networking engine assigns an IP address to the pfSense's first interface using DHCP. The second interface has a private IP address, as this interface is a part of the internal network. The internal network comprises of pfSense 2.1.X, Ubuntu 13.10 and Windows 8. Note that, the VMs can only access the Internet through the firewall (pfSense), as it also has an interface with 'NAT' mode enabled (apart) from an interface with 'Internal Network' mode enabled), which provides the route to the Internet.

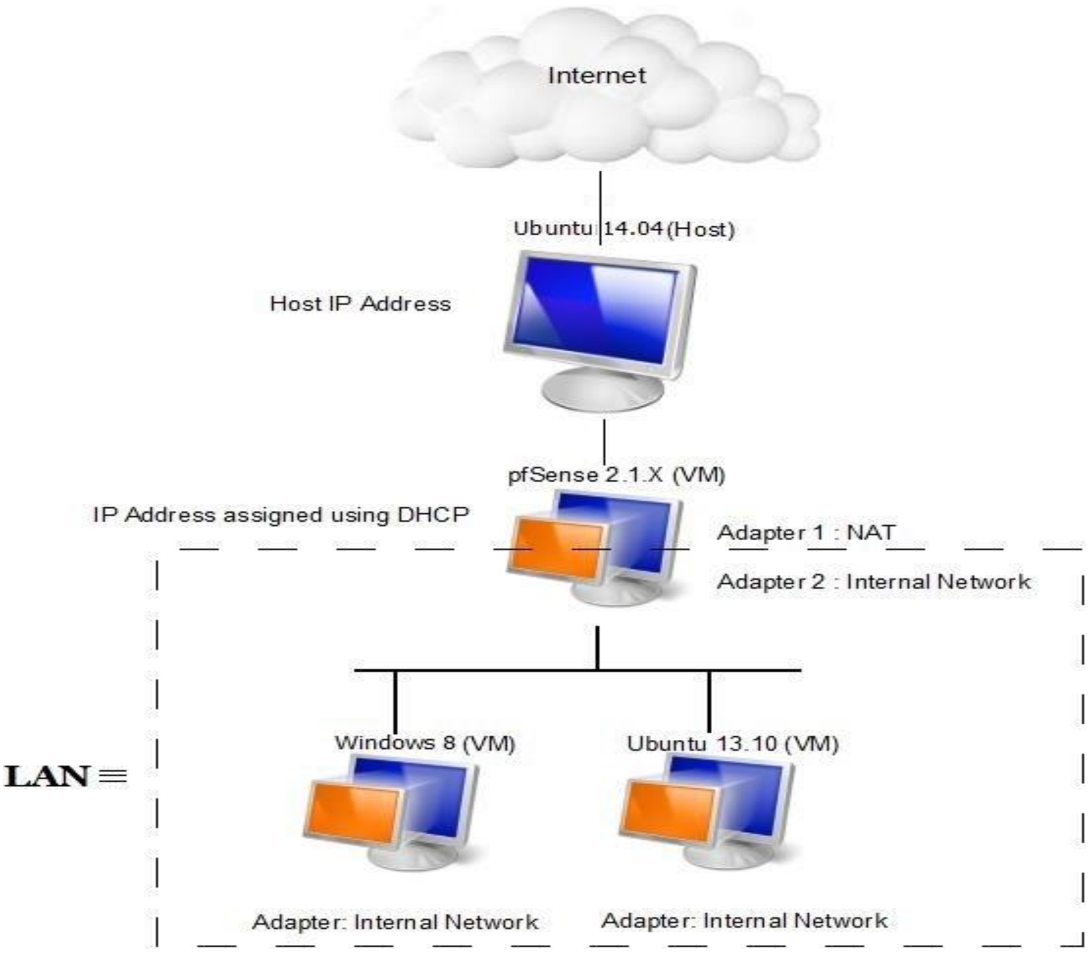


Fig 2: Host Machine and the Virtual Machines as present in the Test Bed

IV. Implementation Of The Solution Using Pfsense

4.1 pfSense

pfSense is a FreeBSD based open source firewall software distribution. A firewall can be made by installing it on a computer system. pfSense firewalls are known to offer high reliability and high-availability. In fact, pfSense offers features that are usually found only in expensive firewalls.[3] Another advantage is that, it can be configured and managed through its user-friendly web interface, thereby obviating the need to have any prior experience with FreeBSD or GNU/Linux Systems. Furthermore, with pfSense, many additional packages like Squid3, SquidGuard3 etc. are available for installation, thereby making it suitable for multifarious applications. [14]

4.2 Setting up a Captive Portal using pfSense

We created a captive portal for authentication using pfSense's web interface. pfSense's captive portal functionality includes several options that facilitate the creation of a feature-rich captive portal. Some of its key features are:

- It allows the management of user groups for captive portal login.
- It supports several types of authentication methods (including RADIUS), we have, however, used Local User Authentication in the test setup for simplicity.
- It allows creation of own captive portal page and error page if the user doesn't want to use the default ones.
- Logout pop up window can also be enabled which will allow users to log themselves out of the captive portal according to their wish.

- Concurrent logins can be disabled so that multiple users cannot log in using the same username and password and use the Internet simultaneously.
- It has a variable called $PORTAL_REDIURL$ which can be set to a URL that all users will be redirected to, after successful login.

4.3 Setting up a Transparent Squid Proxy Server

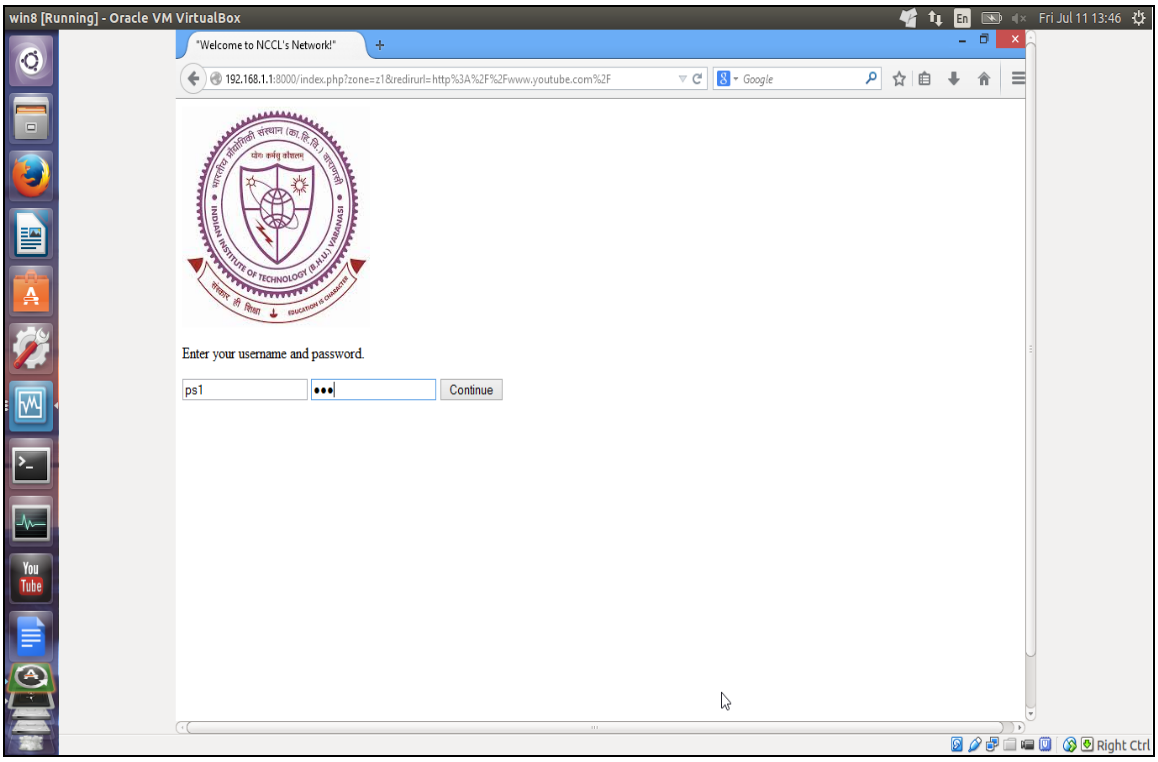
Squid is a high-performance proxy caching server. Its uses include speeding up a web server by caching frequent requests, caching web and DNS lookups, and filtering traffic for security considerations. [4] To use the Squid3 package, it has to be installed separately using the web interface of pfSense. We set up a transparent proxy server using the Squid3 package. Squid3 package also includes many useful features, most important of which is the ability to maintain access logs and cache logs.

4.4 Setting up Squid Guard (Proxy filter)

Squid Guard is an open source URL redirector that is used in conjunction with Squid3 to meet the web filtering requirements. [15] Just like Squid3, SquidGuard3 also needs to be installed separately prior to its use. Standard blacklists are also available for use with SquidGuard, which come with predefined website categories. We used Shalla's blacklists for our test. [16] Using this blacklist, we denied access to the social networking category.

V. TEST

After setting up a captive portal, Squid transparent proxy and SquidGuard proxy filter, the previously developed test bed was used for the final test. Windows 8 VM was made to run and a web browser was opened. The web browser displayed a captive portal page asking for credentials, as shown in the following screenshot.



win8 [Running] - Oracle VM VirtualBox

"Welcome to NCCL's Network!"

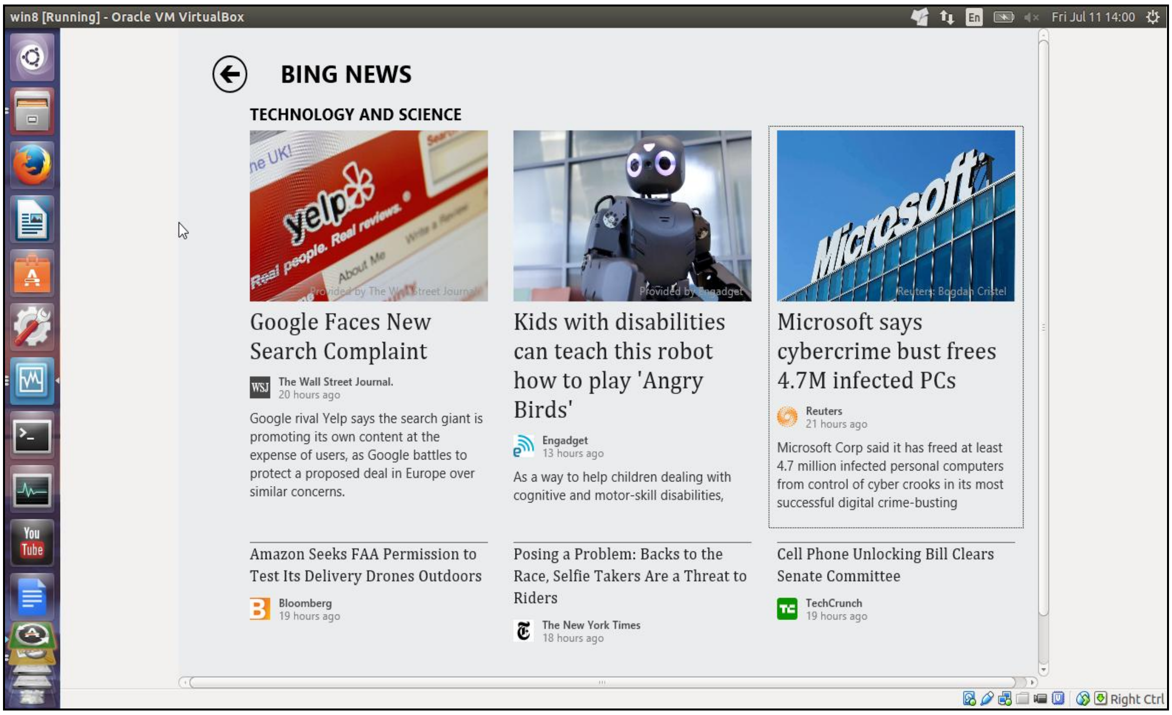
192.168.1.1:8000/index.php?zone=z1&redirurl=http%3A%2F%2Fwww.youtube.com%2F

Enter your username and password.

ps1 ••• Continue

Fig 3: A screenshot showing the captive portal page

After successful login, the news app was tested and it worked, as shown in the following screenshot.



BING NEWS

TECHNOLOGY AND SCIENCE

Google Faces New Search Complaint

The Wall Street Journal. 20 hours ago

Google rival Yelp says the search giant is promoting its own content at the expense of users, as Google battles to protect a proposed deal in Europe over similar concerns.

Kids with disabilities can teach this robot how to play 'Angry Birds'

Engadget 13 hours ago

As a way to help children dealing with cognitive and motor-skill disabilities,

Microsoft says cybercrime bust frees 4.7M infected PCs

Reuters 21 hours ago

Microsoft Corp said it has freed at least 4.7 million infected personal computers from control of cyber crooks in its most successful digital crime-busting

Amazon Seeks FAA Permission to Test Its Delivery Drones Outdoors

Bloomberg 19 hours ago

Posing a Problem: Backs to the Race, Selfie Takers Are a Threat to Riders

The New York Times 18 hours ago

Cell Phone Unlocking Bill Clears Senate Committee

TechCrunch 19 hours ago

Fig 4: A screenshot of the Windows 8 ‘News’ native app working

After testing Windows 8 apps, Ubuntu 13.10 VM was made to run. A web browser was opened which displayed a captive portal page. After successful login, the web browser was closed and the ‘Amazon’ app was launched from the launcher and it worked properly as well. This is shown in the screenshot that follows. (Note that the shown web browser window was opened automatically after launching the app)

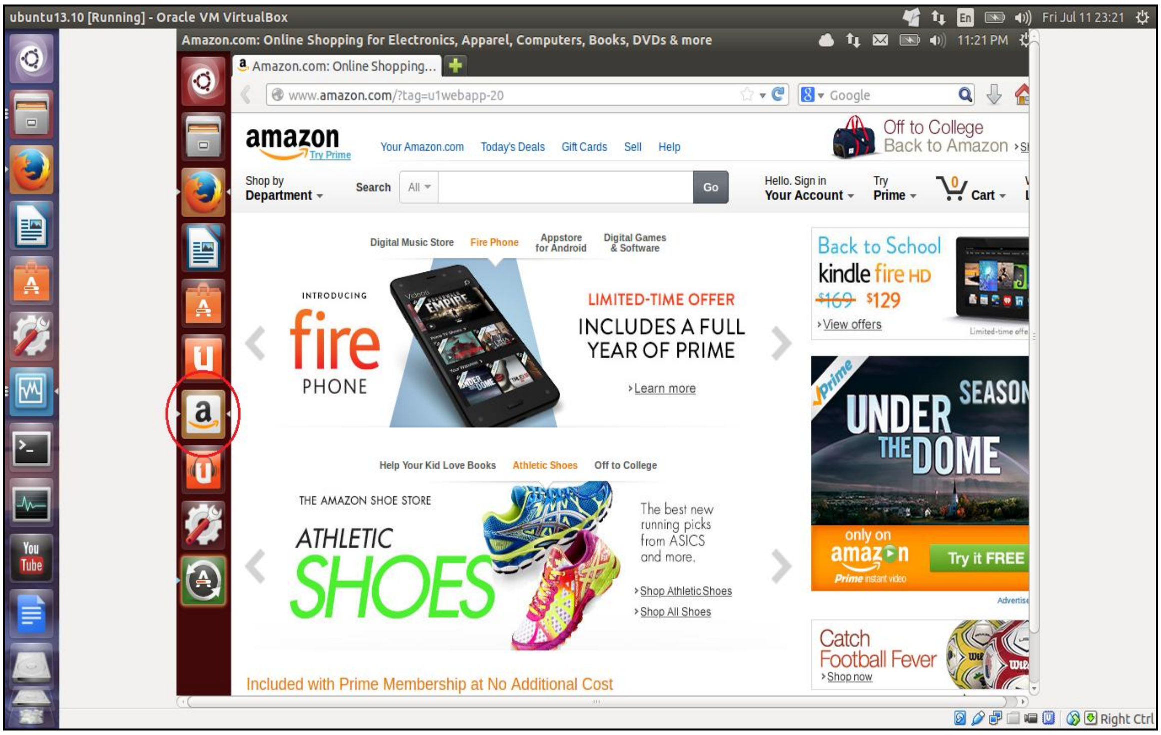
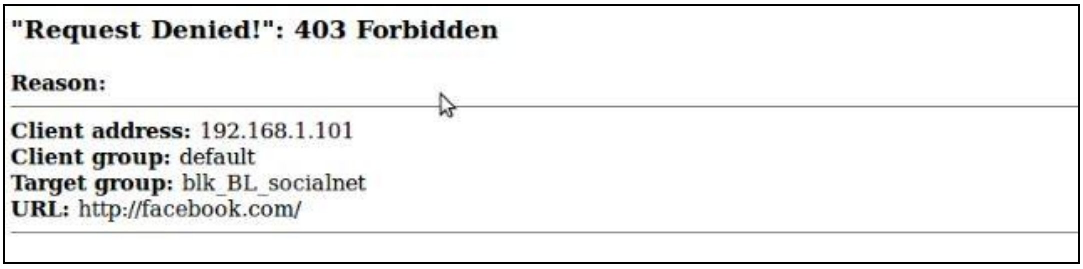


Fig 5: A screenshot of the Ubuntu 13.10 ‘Amazon’ app working. The red mark is to draw attention to the fact that the ‘Amazon’ app is active and that the web browser is opened by this app and not by the user.

To test the functioning of proxy filter, Facebook's website was opened but an error message got displayed, indicating the proper functioning of web filtering functionality of the proxy server. This is shown in the screenshot below.



"Request Denied!": 403 Forbidden

Reason:

Client address: 192.168.1.101
Client group: default
Target group: blk_BL_socialnet
URL: http://facebook.com/

Fig 6: A screenshot of the error page displayed when trying to access blocked web sites

VI. Deployment Scenarios

As apps have become more prevalent and more robust, small and medium enterprises have started relying on them to help themselves grow faster. There are millions of apps that satisfy a wide array of user requirements, including the most trivial ones. However, some of them can play an instrumental role in a company's success. Some of the apps that are particularly important for small/medium enterprises are RightSignature, Geckoboard, Google Drive, Google Analytics, TeamViewer, Asana, LocalVox, Hightail, LinkedIn's Cardmunch, CloudOn, Mint and Square. [17][18]

Furthermore, the education systems are shifting to more modern techniques of learning that require extensive use of apps. In fact, Stanford University has its own mobile learning platform, SMILE (Stanford Mobile Inquiry-based Learning Environment) [19] and some universities (including Boston University) use an LMS (Learning Management System) called Blackboard Learn [20] to provide a powerful, interactive, multimedia-learning environment.

With these advancements, it becomes all the more necessary for the enterprises/ institutions themselves to look for solutions to get the apps working, if their existing network doesn't provide an uninterrupted path (out to the Internet) to the apps. Generally, to get the apps to work properly, an enterprise/educational institute may avoid using a proxy server itself which implies losing all the services provided by it. The use of our proposed solution will ensure that all the apps work properly without sacrificing any web filtering or/and caching in the enterprise/ institute network, since the solution requires using a transparent proxy.

A network diagram for the deployment of the suggested solution in an enterprise/ institute is given below. (Note that the captive portal functionality is assumed to be built into the firewall)

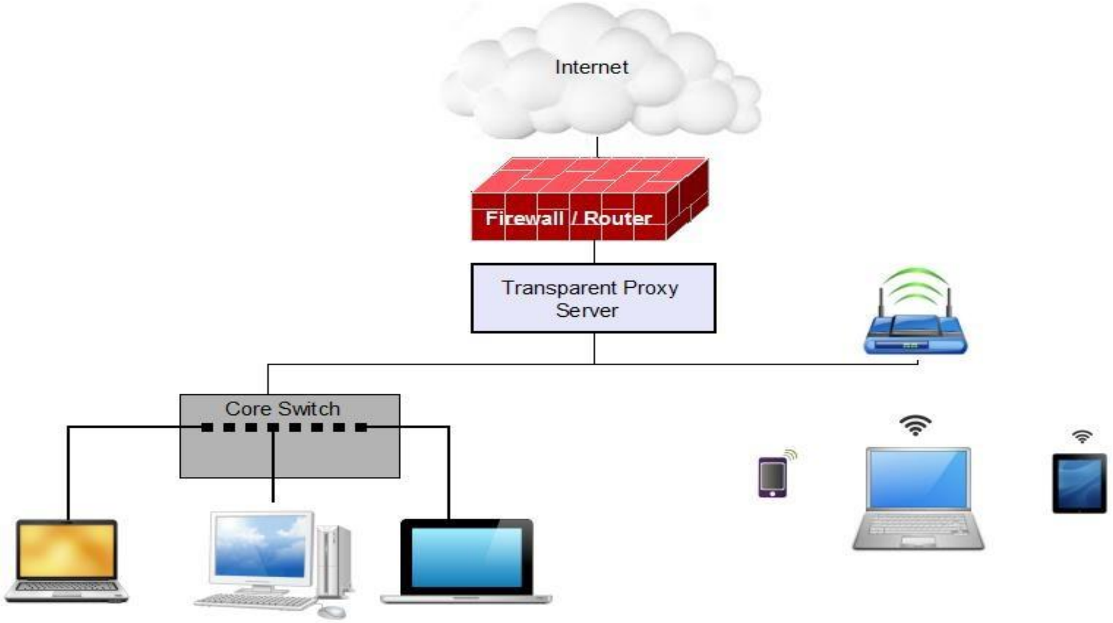


Fig 7: A General network diagram for the deployment of the proposed solution

VII. Conclusion

In this paper, we have suggested a solution to get the apps to work without having to avoid the use of a proxy server itself, since it can serve several important purposes like caching, web filtering etc. Our proposal employs a transparent proxy in conjunction with a captive portal. A captive portal has been used as the transparent proxy cannot authenticate the users on its own. We finally demonstrated that, using the proposed solution, Windows 8 native apps and Ubuntu 13.10 apps worked well. We also showed that the web filtering was taking place, indicating the presence of a proxy server.

REFERENCES

[1] "Transparent vs Direct Proxies – Netbox Blue" http://netboxblue.com/sites/2012.netboxblue.com/files /Application%20Brief%20%20Direct%20vs%20Transparent%20Proxy-v1.1-Nov13.pdf

[2] "VirtualBox" - https://www.VirtualBox.org/wiki/VirtualBox

[3] S.Miller, "Configure a professional firewall using pfSense", *Free Software Magazine*.

[4] "Squid FAQ: About Squid".- http://wiki.squid-cache.org/SquidFaq/AboutSquid

[5] [RFC2616] Fielding, R., Gettys, J., Mogul, J., Frystyk, H., Masinter, L., Leach, P., and T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999. (http://www.ietf.org/rfc/rfc2616.txt)

[6] [RFC1919] Chatel, M., "Classical versus Transparent IP Proxies", RFC 1919, March 1996. (http://www.ietf.org/rfc/rfc1919.txt)

[7] R.Auger, "Socket Capable Browser Plugins Result In Transparent Proxy Abuse", *The Security Practice*.

[8] Ozgur Ercetin, *Market- Based Resource Allocation for Content Delivery in the Internet*, doctoral diss., University of Maryland, College Park, United States, 2002

[9] Jaume Barceló, Miquel Oliver, and Jorge Infante, "*Adapting a Captive Portal to Enable SMS-Based Micropayment for Wireless Internet Access", Lecture Notes in Computer Science Volume 4033*, 2006, 78 – 89

[10] Rustam Jemurzinov, *Authentication and authorization service for a community network*, Master's Thesis, Lappeenranta University of Technology, Lappeenranta, Finland, 2008.

[11] p 11 Chapter 1, p 94 Chapter 6, p 99 Chapter 6, "Oracle VM VirtualBox User Manual" - http://download.VirtualBox.org/VirtualBox/UserManual.pdf

[12] "Virtualization in education"- IBM, http://www7.ibm.com/solutions/in/education/download/Virtualization%20in%20Education.pdf

[13] "Network configuration in VirtualBox" http://www.thomas-krenn.com/en/wiki/Network_Configuration_in_VirtualBox

[14] V. Danen, "DIY pfSense firewall system beats others for features, reliability, and security", *TechRepublic*.

[15] "SquidGuard" - http://www.Squidguard.org/

[16] "Shalla's Blacklists" - http://www.shallalist.de/

[17] "10 brilliant apps small businesses should use" - http://www.forbes.com/sites/ilyapozin/2012/05/29/10-brilliant-apps-small-businesses-should-use/

[18] "10 Essential tablet apps for business" - http://mashable.com/2012/08/15/tablet-mobile-apps-business/

[19] S. Seol, A. Sharp, & P. Kim, Stanford Mobile Inquiry-based Learning Environment (SMILE): using mobile phones to promote student inquires in the elementary classroom, *Proc. 2011 International Conference on Frontiers in Education: Computer Science & Computer Engineering*, 270-276.

[20] "Blackboard Learn" - http://www.bu.edu/tech/services/teaching/lms/blackboard/