**Project 0x83E**

**Great White Buffalo**

**ITIS 2110-L04**

**11-30-2012**

Samuel Stanton, Phillip Lopez, Michael Martin,

Josette Seitz, Juan Davilla

**No table of figures entries found.**

Table of Contents

[Table of Figures 3](#_Toc441194909)

[Table of Tables 4](#_Toc441194910)

[Overview 5](#_Toc441194911)

[Objectives 5](#_Toc441194912)

[Network Diagram 6](#_Toc441194913)

[Setting up the Local Network 6](#_Toc441194914)

[End Devices 6](#_Toc441194915)

[Intermediary Devices 8](#_Toc441194916)

[Segmenting the Network 8](#_Toc441194917)

[Subnetting 9](#_Toc441194918)

[Original Plan 9](#_Toc441194919)

[Making Room for Expansion 9](#_Toc441194920)

[Addressing the devices 9](#_Toc441194921)

[Setting up the “Internet” 10](#_Toc441194922)

[Linking to the “ISP” 10](#_Toc441194923)

[An Untrusted Host 10](#_Toc441194924)

[An External Web Server 10](#_Toc441194925)

[Communicating with the Internet 11](#_Toc441194926)

[Securing the Network 11](#_Toc441194927)

[The Access Control Lists 11](#_Toc441194928)

[Problems with Original ACLs 12](#_Toc441194929)

[Making the ACLs Work 12](#_Toc441194930)

[Final ACLs 13](#_Toc441194931)

[Documented Success 14](#_Toc441194932)

[Connecting to the External Web Server from Inside 14](#_Toc441194933)

[Connecting to the Internal Web Server from Outside 15](#_Toc441194934)

[Scanning for Open Ports 16](#_Toc441194935)

[Summary 17](#_Toc441194936)

[Appendix 18](#_Toc441194937)

[Windows Server 2008 Setup 18](#_Toc441194938)

[Installing Active Directory and DNS 18](#_Toc441194939)

[Adding a user to Active Directory 23](#_Toc441194940)

[Creating an A Record 27](#_Toc441194941)

[Database Server Setup 30](#_Toc441194942)

[Web Server Setup 30](#_Toc441194943)

[Installing Apache and PHP 30](#_Toc441194944)

[Connecting Windows 7 Client to domain 31](#_Toc441194945)

[Cisco 1841 Router Configuration 35](#_Toc441194946)

[Base Configuration 35](#_Toc441194947)

[DHCP 36](#_Toc441194948)

[Access Control Lists 39](#_Toc441194949)

[Cisco Catalyst 2950 Switch Configuration 40](#_Toc441194950)

# Table of Figures

Figure 1 6

Figure 2 14

Figure 3 15

Figure 4 16

Figure 5 18

Figure 6 19

Figure 7 19

Figure 8 20

Figure 9 20

Figure 10 21

Figure 11 22

Figure 12 23

Figure 13 24

Figure 14 24

Figure 15 25

Figure 16 26

Figure 17 27

Figure 18 28

Figure 19 31

Figure 20 31

Figure 21 32

Figure 22 33

Figure 23 33

Figure 24 34

# Table of Tables

Table 1 7

Table 2 9

Table 3 9

Table 4 11

Table 5 13

Table 6 13

Table 7 13

Table 8 14

Table 9 31

Table 10 31

Table 11 36

Table 12 36

Table 13 37

Table 14 37

Table 15 38

Table 16 38

Table 17 38

Table 18 39

Table 19 40

Table 20 41

Table 21 41

Table 22 42

Table 23 42

**Project Overview and Objectives**

# Overview

For this Final Project we chose to implement and small network, similar to that which may be found in a small business environment. Using topics learned in lecture and through outside research, we implemented a such a network consisted of multiple subnets, a DMZ hosting an HTTP server and a Secure subnet hosting business critical services including DNS, authentication, and database services. We configured a Cisco infrastructure running NAT, DHCP, and access control lists to handle security. We were able to implement the network using lab computer running virtual machines and networking equipment from the lab. As an added bonus we setup second small network to simulate the untrusted Internet and to test the configuration of our access control lists. The following documents our steps in configuring the network, explains issues that arose during the configuration, and demonstrates the finished projects using screen shots and other examples.

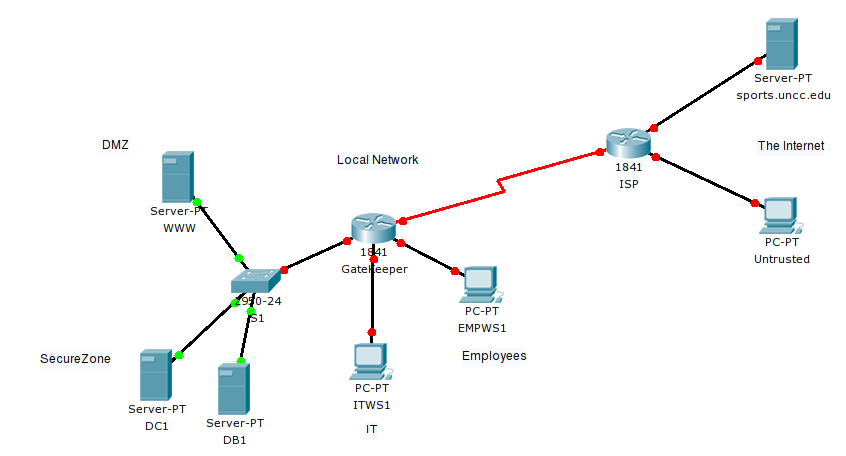
# Objectives

The following is a list of our main objectives and goals upon completion of the project:

* Two subnets that represent organizational departments. IT and Employees
* Demilitarized Zone (DMZ) to host services accessible from the Internet
  + HTTP Server on the DMZ
  + HTTP server should run PHP and pull content from a MySQL database on a different subnet
* Secure Zone hosting internal services, accessible only but the local subnets
  + Windows Server 2008 device running Active Directory for authentication and DNS
  + Debian Server running MySQL database, accessible only by the HTTP server to display content
* Implement separate network to simulated the untrusted Internet
  + One subnet containing an “untrusted” client device
  + Second subnet containing a public Web server
* Configure DHCP on the local Cisco router for automated IP address assignment
* Configure NAT on the local Cisco router to allow local clients to access devices outside the local network
* Configure access control lists to prohibit unwanted traffic from entering the local network

Upon completion of the project it is our intent to accomplish and demonstrate the objectives listed above.

## Network Diagram



Figure

# Setting up the Local Network

The section goes over the planning and implementation of the initial network. First we will discuss the typed of end and intermediary devices that will be included in the network and then we will go over how we decided to segment our network, both physically and logically.

## End Devices

To meet our objectives we needed to include devices to represent each department subnet as well as devices to provide specific services, DNS, authentication, database, and HTTP.

To represent the two department subnets, we one lab computer for each subnet. We installed a Window 7 virtual machine on each physical lab computer to act as a client device on each subnet. One subnet is named Employees, and this is where regular employees of our organization will reside. The second subnet, named IT, is where IT department personal of our organization will reside. The name of the client machine representing Employees is EMPWS1 and the name of the client machine representing the IT is ITWS1.

To provide for centralized authentication on the network so that each employee would not require a user name on every end device they may use, we set up a Windows Server 2008 virtual machine with the Active Directory role installed. This provides for a single database of users in which any end device on our network can communicate with to authenticate a user. We also installed the DNS role on this machine to provide for domain name to IP mapping.

We configured a second virtual machine, this one running the Debian operating system to host our database server. We decided to use the MySQL database server to provide database services as it is freely available and easy to set up. Since no end users would be using this machine directly, except for IT personnel for maintenance purposes, we have excluded the GUI interface and other non-essential services.

Both the Windows Server 2008 Virtual Machine (VM) and the Debian VM running MySQL will reside on a third physical lab computer, together representing the SecureZone subnet. This subnet will only be accessible from with the local network and even that access will be restricted as appropriate. The Windows Server 2008 VM will be named DC1 and the Debian VM will be named DB1

Finally, we setup a fourth physical lab computer hosting another Debian virtual machine. On this virtual machine, we installed the Apache Web Server and PHP 5 to host a dynamic web site that pulls content from the MySQL database server on the SecureZone subnet. We also excluded the GUI from this VM as it is unnecessary and only IT personnel will be accessing this machine for maintenance. This Debian VM will be named WWW. The final lab computer will represent our fourth subnet, the DMZ.

Table 1 provides a table of the host devices, including their host name, subnet, how their IP addresses will be configured (static or dynamic), and a brief description of their roles.

#### Table 1

| Host name | Subnet | IP Configuration | Description |
| --- | --- | --- | --- |
| EMPWS1 | Employees | Dynamic | Represents Employees subnet |
| ITWS1 | IT | Dynamic | Represents IT subnet |
| DC1 | SecureZone | Static | Provides DNS and authentication services |
| DB1 | SecureZone | Static | Provides database services to the HTTP server |
| WWW | DMZ | Static | Provides HTTP services to local network and the Internet |

Details on how we installed and configured Active Directory, DNS, MySQL, and Apache are covered in the Appendix as well as the process of adding the client devices to the domain.

## Intermediary Devices

To connect the devices together so that they can communicate we needed two intermediary devices. A router, to handle routing traffic between subnets and the Internet, and a switch to provide VLANs because of limitations of lab equipment.

We used one of the labs Cisco 1841, 4 port routers to handle routing traffic between the different subnets and the Internet and provide DHCP services to each department subnet for easy addition of client devices to the network.. The router will also be configure with Access Control Lists (ACLs) to control the type of traffic that is allowed in and out of the network and to where that traffic is allowed once inside the network. We have named the router GateKeeper.

We decided to use a Cisco 2950, 24 port switch to provide for our VLAN configuration, named S1. The switch will host two VLANs, including the DMZ subnet and the SecureZone subnet. The switch will be named S1

The router interfaces will be configured as follows:

Table

|  |
| --- |
| * FastEthernet 0/0 will be the gateway for the Employees subnet * FastEthernet 0/1 will be the gateway for the IT subnet * FastEthernet 0/1/0 will be split into to sub-interfaces, FastEthernet 0/1/0.2 and FastEthernet 0/1/0.3   + FastEthernet 0/1/0.2 will be the gateway for the SecureZone subnet   + FastEthernet 0/1/0.3 will be the gateway for the DMZ subnet * FastEthernet 0/1/1 will be the link to an ISP router providing internet access to the local network |

The switch will be configured as follows:

Table

|  |
| --- |
| * FastEthernet 0/1 – 0/8 will be part of the SecureZone VLAN * FastEthernet 0/9 – 0/16 will be part of the DMZ subnet * FastEthernet 0/24 will be the trunk port to the routers FastEthernet 0/1/0 interfaces * FastEthernet 0/10 – 0/23 will be reserved for future expansion |

Details on how the routers and switch are configured can be found in the Appendix.

## Segmenting the Network

In order to provide for simplified management and security, we have segmenting the network by department and use. The following part of this section discusses the logical segmentation of the local network.

### Subnetting

Because of the small size of our network, we decided to go with a 192.168.1.0 /24, Class C address space. The host requirements for each subnet are:

* Employees – No more than 30 hosts
* IT – No more than 30 hosts
* SecureZone – Currently up to 6 hosts, may double in the future
* DMZ – Currently up to 6 hosts, may double in the future

### Original Plan

The following was our original subnetting scheme:

* Employees – 192.168.1.0 /27
* IT – 192.168.1.32 /27
* SecureZone – 192.168.1.64 /29
* DMZ – 192.168.1.72 /29

This configuration allows for the maximum number of hosts on each department subnet and the current maximum number of hosts on the DMZ and SecureZone subnets. However, if expansion were to occur, such as new department subnets added or more devices added to the SecureZone and DMZ subnets, we would have to completely redesign our subnetting schem

### Making Room for Expansion

To provide for a much easier expansion process on the network, we redesigned our subnetting scheme as follows:

* Employees – 192.168.1.0 /27
* IT – 192.168.1.32 /27
* SecureZone – 192.168.1.224 / 29
* DMZ – 192.168.1.240 /29
* 192.168.1.64 – 192.168.1.223 Open for expansion.
* 192.168.1.232 /29 open for expansion of the SecureZone subnet or addition of a new subnet if needed
* 192.168.1.248 /29 open for expansion of the SecureZone subnet or addition of a new subnet if needed

The above subnetting scheme is more efficient than our original scheme because if a new subnet is added to the network, we can simply select one of the unused address ranges in the middle to provide addressing to the new subnet. Also, the SecureZone and the DMZ subnets can easily be expanded to include the unused addresses in between them by changing the subnet mask fro /29 to /27, doubling the number of addresses available on each subnet.

### Addressing the devices

On each subnet, the gateway will be statically assigned the last IP address on the subnet. End device addressing will begin with the first available address on each subnet, On the department subnets, the last 5 addresses will be reserved for static devices such as printers and all other addresses will be assigned via DHCP from the router. On the DMZ and SecureZone subnets, all devices will be statically assigned.

Table 4 shows the IP address assignment across the network, including each device name, IP address, and subnet mask.

Table

| Device | IP Address | Subnet Mask |
| --- | --- | --- |
| GateKeeper Fa0/0 | 192.168.1.30 | 255.255.255.224 |
| GateKeeper Fa0/1 | 192.168.1.62 | 255.255.255.224 |
| GateKeeper Fa0/1/0.2 | 192.168.1.230 | 255.255.255.249 |
| GateKeeper Fa0/1/0.3 | 192.168.1.246 | 255.255.255.249 |
| EMPWS1 | Dynamic (192.168.1.1-25) | 255.255.255.224 |
| ITWS1 | Dynamic (192.168.1.33-57) | 255.255.255.224 |
| DC1 | 192.168.1.225 | 255.255.255.249 |
| DB1 | 192.168.1.226 | 255.255.255.249 |
| WWW | 192.168.1.241 | 255.255.255.249 |

# Setting up the “Internet”

This section briefly covers the implementation of a second network to simulate the untrusted Internet. This network will consist of a second Cisco 1841 router to create a link between our local network and the Internet. Connected to the router will be two subnets. One consisting of an untrusted host or client computer and the other consisting of a publicly accessible web server. The router will be named ISP as it represents the organizations link to the Internet.

## Linking to the “ISP”

The FastEthernet 0/1/0 interface of the ISP router will be used to link to the WAN interface (FastEthernet 0/1/1) on GateKeeper. This network that connects the local network to the rest of the internet will reside on the 174.110.255.0 /24 address space. 174.110.255.253 will be assigned to the ISP interface and 174.110.255.254 will be assigned to the GateKeeper interface. Other addresses on this space represent those assigned by the ISP to other customers and two of them will be used for NAT, discussed later in this section.

## An Untrusted Host

To represent the untrusted host, we installed a Debian VM on another lab computer. This host is connected to the FastEthernet 0/0 interface of the ISP router. We have installed Nmap on the untrusted host to test the ACLs implemented on the GateKeeper router.

The untrusted host will reside on the 174.110.161.0 /24 address space. The 174.110.161.30 address will be statically assigned to the VM and 174.110.161.254 will be assigned to the router interface

## An External Web Server

The external web server will also be a Debian VM installed on a lab computer. The VM will have no GUI and be running Apache to display a static web page to visitors. The VM will be used to test connectivity and ACLs for outbound traffic from the local network. This web server will reside on the 198.133.219.0 /24 address space and with be statically assigned to 198.133.219.25. The router interface will be assigned 198.133.219.254. Finally, the DNS name assigned to the public web server will be sports.uncc.edu.

## Communicating with the Internet

To allow the clients inside the local network to communicate with devices on the Internet, we needed to configure Network Address Translation, or NAT, to allow private IP addresses to map to public IP addresses. On the GateKeeper router, we configure a one-to-many NAT, to allow client devices on the department subnets to communicate with devices on the Internet, particularly web services. We also configured a one-to-one NAT for the internal web server to allow host on the Internet communicate with our web server.

For the one-to-many NAT we configured to the address 174.110.255.50 to map to all of the addresses on the Employees and IT subnets, (192.168.1.1 – 62). The one-to-one NAT configuration for the web server maps 174.110.255.51 to the web servers internal address of 192.168.1.241. We configured static NAT for the web server to allow devices to make requests of the server and not just respond to it.

NAT configuration is detailed in the Router Configuration section of the Appendix.

# Securing the Network

We have taken a few step to secure our local network. The our main focus with securing the network involved the creation of Access Control Lists to dictate what traffic was permitted on our network and to which subnets specific traffic was allow or denied. Along we the ACLs, we placed the Database server on a secure subnet, and configured it to only allow the HTTP server to make requests from the database. Host firewalls are left on for all end devices and only necessary services are running on the servers and end devices.

## The Access Control Lists

We used the following objectives when designing our ACLs:

* Prevent spoofing of reserved IP addresses from entering the local networking
* Allow untrusted devices to communicate with the Apache web server on ports 80 and 443 only
* Allow local client devices to communicate with the Apache server on port 80 and 443
* Allow local client devices to communicate with Web servers on the Internet on ports 80 and 443
* Allow only the Apache server to communicate with the MySQL database on port 3306
* Prevent all other devices from communicating with the MySQL server
* Allow only local department devices to communicate with the Windows Server 2008 machine for authentication
* Allow only local devices to make DNS requests to the Windows Server 2008 machine.
* Allow all devices to ping all other devices except for the MySQL server, in which ICMP should be blocked

## Problems with Original ACLs

When initially configuring our ACLs we ran into a problem. After building the ACLs and assigning them to their specified interfaces, devices on the Internet could no longer communicate with the local Apache web server and department devices could no longer communicate with web server on the Internet. After careful examination of our ACLs, we found that we needed to correct the addresses that our HTTP rules were applied to.

## Making the ACLs Work

The original ACL rules we created to allow Web Servers on the Internet to respond to requests from local department clients are below:

Table

|  |
| --- |
| permit tcp any eq 80 192.168.1.0 0.0.0.63 established  permit tcp any eq 443 192.168.1.0 0.0.0.63 established |

The problem occurs because we are focusing our ACLs on traffic being sent to the internal IP address range of the local clients. This cause a problem for clients requesting web pages on the Internet because these clients are NAT’d at the WAN interface. When an external web server responds to a client it is responding to the address 174.110.255.51 not the clients local 192.168.1.x address. To correct this issue we simply needed to replace the destination address to the NAT’d public address as follows:

Table

|  |
| --- |
| permit tcp any eq 80 host 174.110.255.51 established  permit tcp any eq 443 host 174.110.255.51 established |

The same problem applied to outside devices connecting to our local web server. The original ACL looked like this:

Table

|  |
| --- |
| permit tcp any host 192.168.1.241 eq 80  permit tcp any host 192.168.1.241 eq 443 |

When external clients would make requests to 174.110.255.50, the traffic was blocked because the ACL rules were applied before the public address was mapped back to the local web server IP address. Again, to fix this we simply needed to change the destination address to the public facing IP address of 174.110.255.51.

## Final ACLs

The following is the final ACL configuration for GateKeeper. Lines beginning with and ‘!’ are comments.

Table

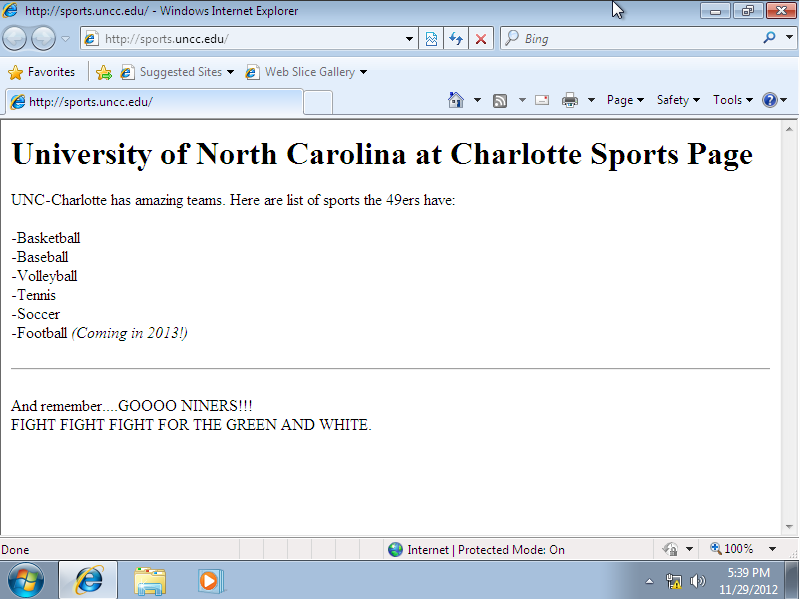
|  |
| --- |
| !Place on WAN interface inbound to router  ip access-list extended UNTRUSTED  permit icmp any any  permit tcp any host 174.110.255.51 eq 80  permit tcp any host 174.110.255.51 eq 443  permit tcp any eq 80 host 174.110.255.50 established  permit tcp any eq 443 host 174.110.255.50 established  !Place on DMZ subinterface inbound to router  ip access-list extended DMZ  permit icmp any any  permit tcp host 192.168.1.241 host 192.168.1.226 eq 3306  permit tcp host 192.168.1.241 eq 80 any established  permit tcp host 192.168.1.241 eq 443 any established  !Place on SecureZone subinterface inbound to router  ip access-list extended SECZONE  permit icmp any any  permit tcp host 192.168.1.226 eq 3306 host 192.168.1.241  permit udp host 192.168.1.225 eq 53 any  !Below is for local AD Authentication  permit ip host 192.168.1.225 192.168.1.0 0.0.0.63 established  !Anti-spoofing placed on WAN interface, both inbound and outbound  ip access-list extended ANTI-SPOOF  deny ip 10.0.0.0 0.255.255.255 any  deny ip 172.16.0.0 0.15.255.255 any  deny ip 192.168.0.0 0.0.255.255 any  deny ip 127.0.0.0 0.255.255.255 any  deny ip 169.254.0.0 0.0.255.255 any  deny ip 100.64.0.0 0.63.255.255 any  deny ip 192.0.2.0 0.0.0.255 any  deny ip 198.18.0.0 0.1.255.255 any  deny ip 198.51.100.0 0.0.0.255 any  deny ip 203.0.113.0 0.0.0.255 any  deny ip 255.255.255.255 0.0.0.0 any  deny ip 192.168.1.0 0.0.0.255 any |

Details on the above ACLs can be found in the Appendix in the router configuration section.

# Documented Success

## Connecting to the External Web Server from Inside

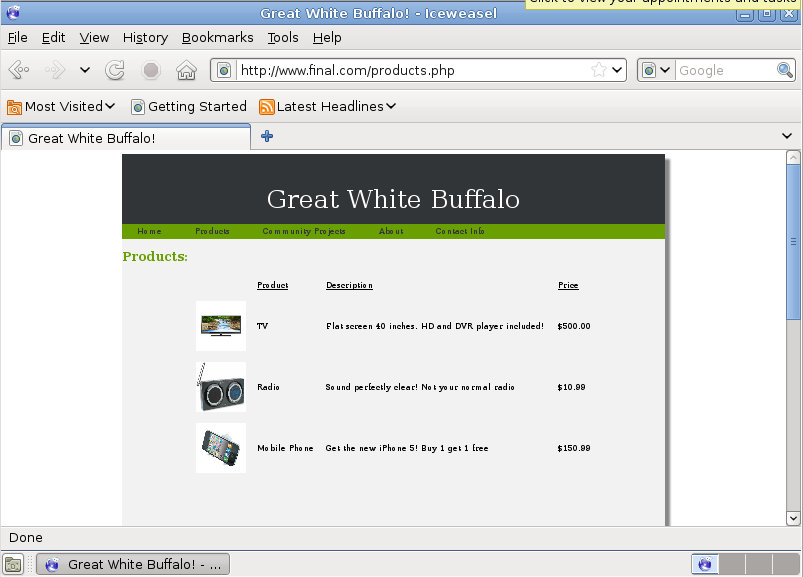
Figure 2 below shows the successful connection to a web server on the Internet by a Windows 7 client on our local network.



Figure

## Connecting to the Internal Web Server from Outside

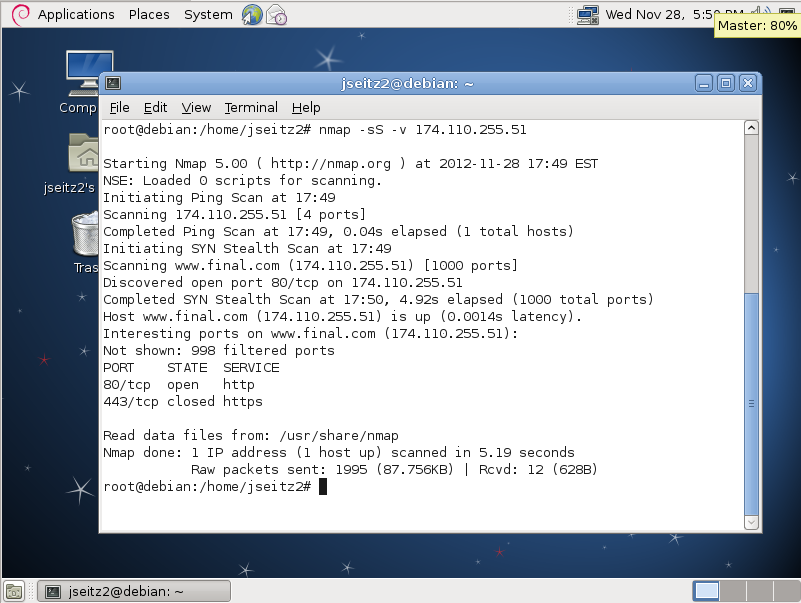
Figure 3 below shows the successful connection to our local web server by the untrusted client on the Internet.



Figure

## Scanning for Open Ports

Figure 4 below shows a Nmap port scan of the local networks public IP addresses. Notice that only port 80 and 443 are found to be open.



Figure

# Summary

To summarize our project, we designed and configured a network, representative of one that might be found in a small business environment. Our network consists of 4 subnets, including 2 department subnets, a Secure Zone and a DMZ. Our DMZ contains a publicly accessible web server that runs Apache, serves dynamic PHP pages and pulls content from a MySQL server running in a separate, secure subnet. This secure subnet, named SecureZone, contains the MySQL database and a Windows Server 2008 machine that provides DNS and authentication services to the local network. Each department subnet has a client device that represents its respective subnet.

We have also implanted a second network to simulate the untrusted Internet. On this network, we configured a “public” Web server that is accessible by any one. We also added an “untrusted” client machine to represent a visitor to our web site and a potential attacker.

To secure our network, we configured Access Control Lists on our router to allow or deny certain traffic and control where permitted traffic could go on our network. When creating the ACLs we ran into some problems that we had to correct to ensure the usability of the network. After making these corrections we had a network that met our access restriction guidelines and allowed end users to access web content on the Internet.

We also used subnetting to segment our network logically. While designing our IP addressing scheme, we re-worked how addresses were assigned to make future expansion of the network more efficient for network administrators.

Overall, our project, as planned was successful, requiring some thinking and troubleshooting.

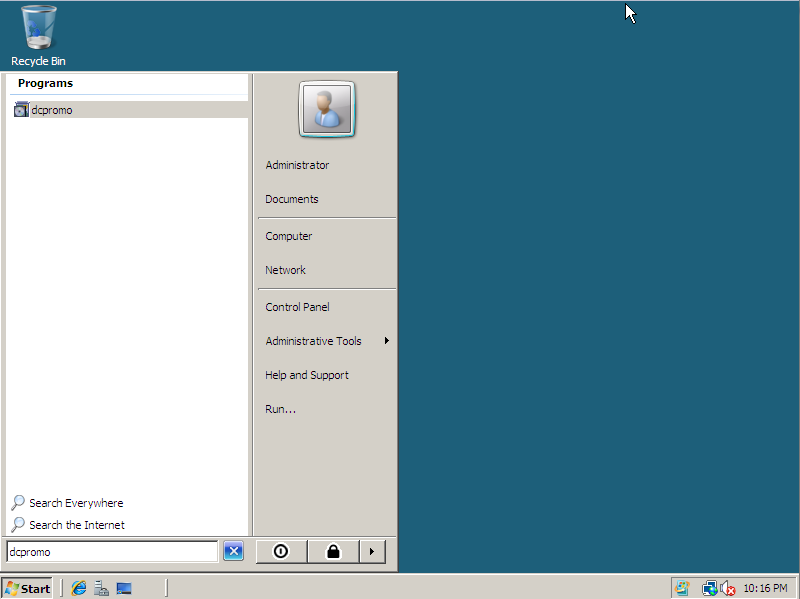
# Appendix

## Windows Server 2008 Setup

### Installing Active Directory and DNS

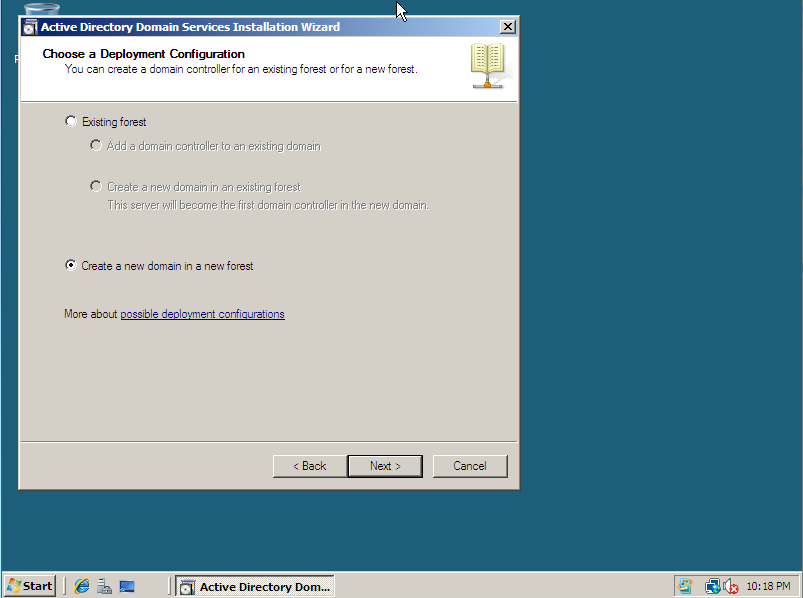
The following steps were used to install Active Directory

1. Open the Start menu, type “dcpromo” in the search box and press Enter.



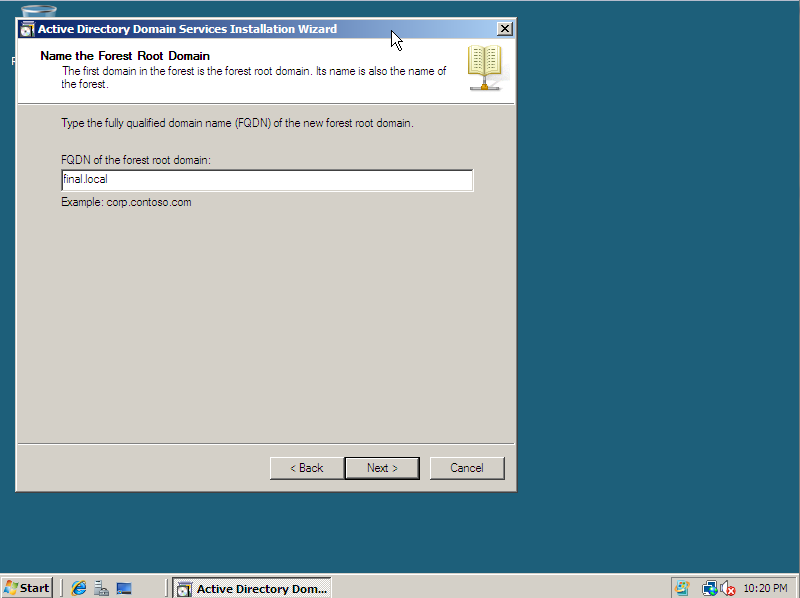
Figure

1. Click Next at the first two screens.
2. Select the “Create a new domain in a new forest radio button and click next.



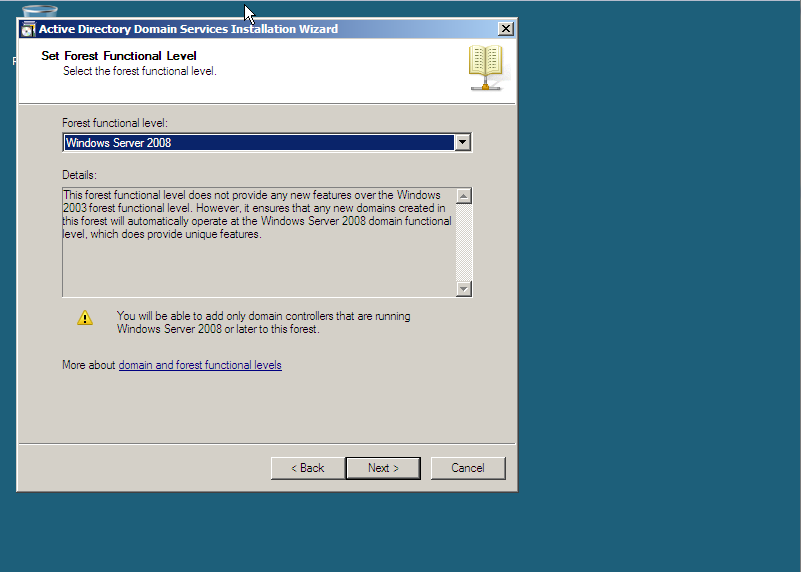
Figure

1. Enter the full qualified domain name in the text box and click next.



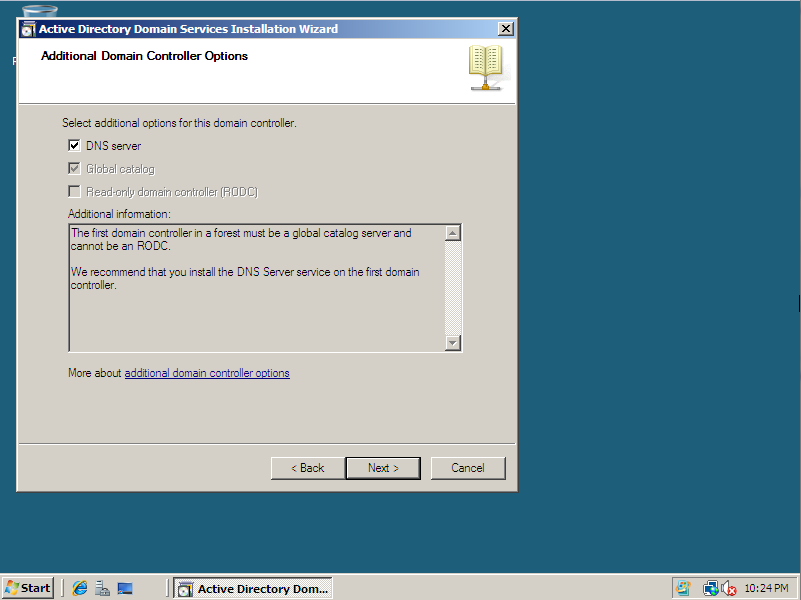
Figure

1. In the “Forest functional level” drop down menu select Windows Server 2008 and click next.



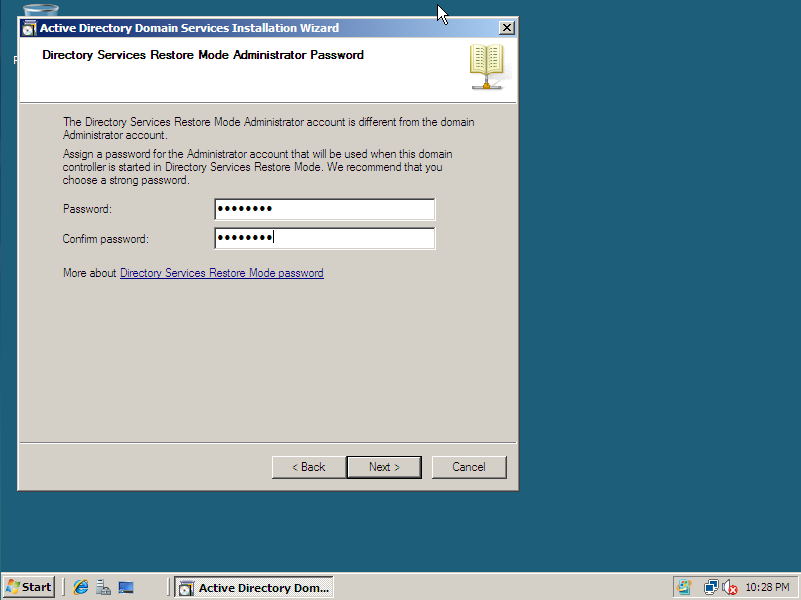
Figure

1. After the wizard checks the DNS configuration, ensure that the DNS Server check box is selected and click next. This will install DNS services.



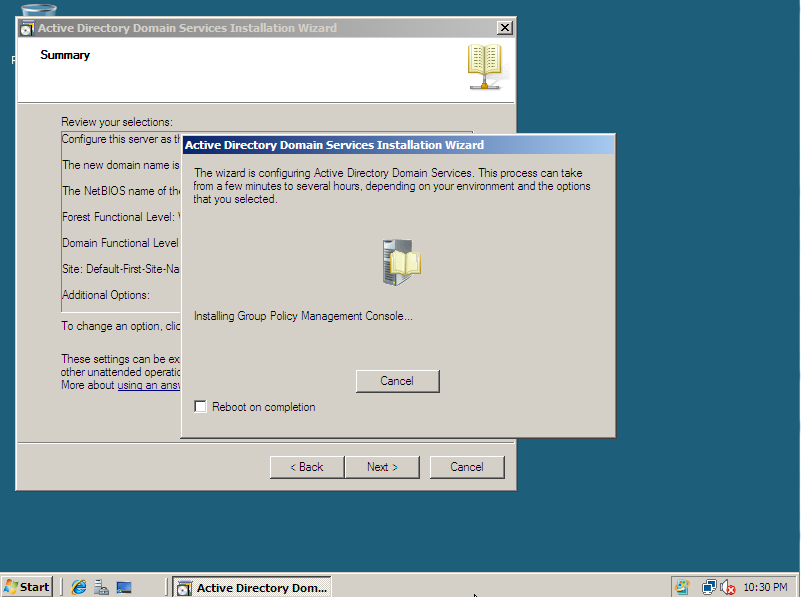
Figure

1. Leave the deafults on the next page and click next
2. Enter the Administrator password and click next



Figure

1. Click next on the Summary screen and wait for the DNS installation to complete



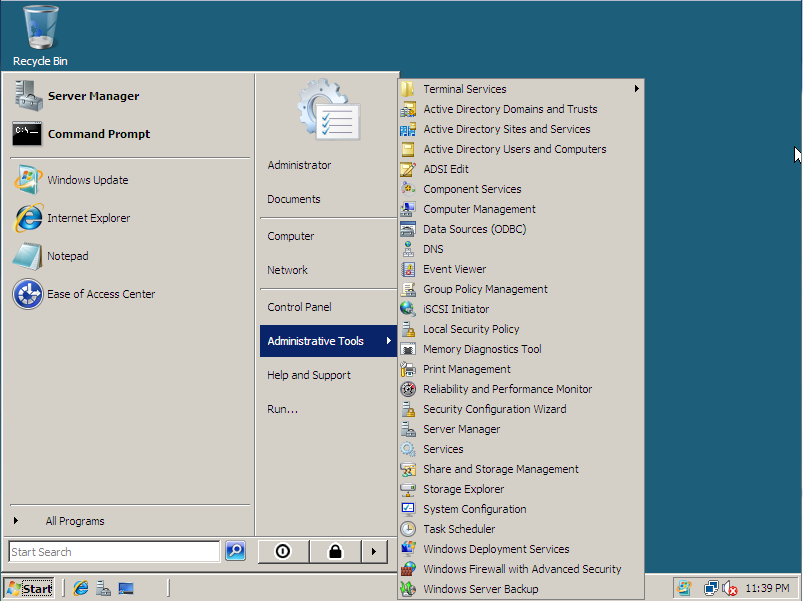
Figure

1. Finally, click the Finish button and reboot the server to complete the installation.

### Adding a user to Active Directory

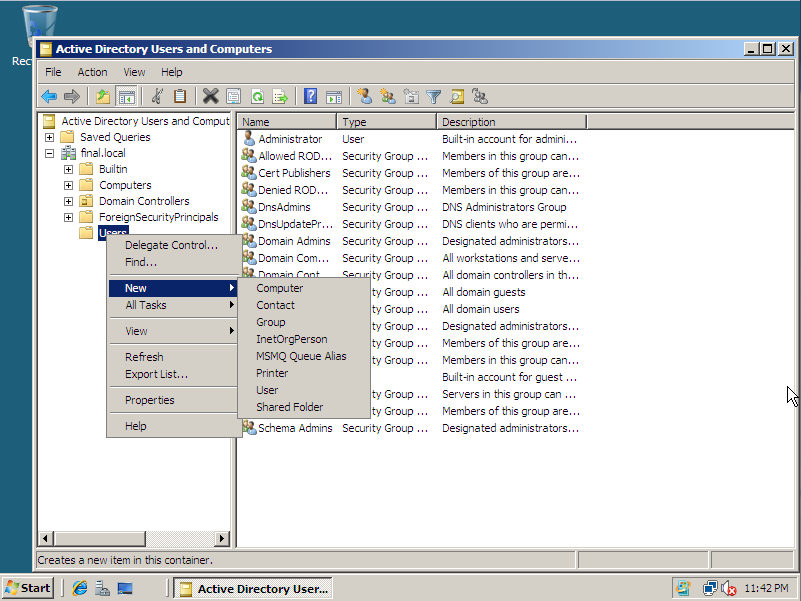
The following steps add a user to the Active Directory Database

1. Open the Start Menu and Select Administrative Tools > Active Directory Users and Computer



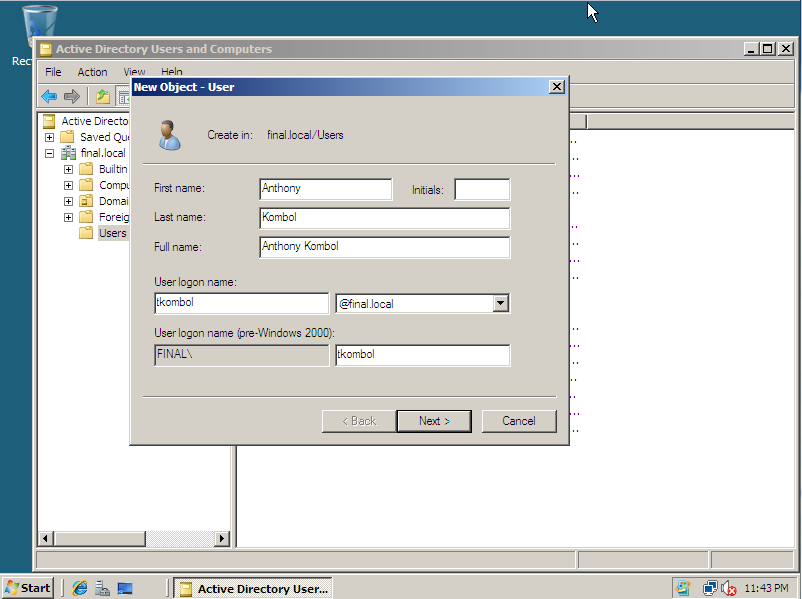
Figure

1. In the left pane of the window expand final.local.
2. Right click the “Users” Organizational Unit and select New > User



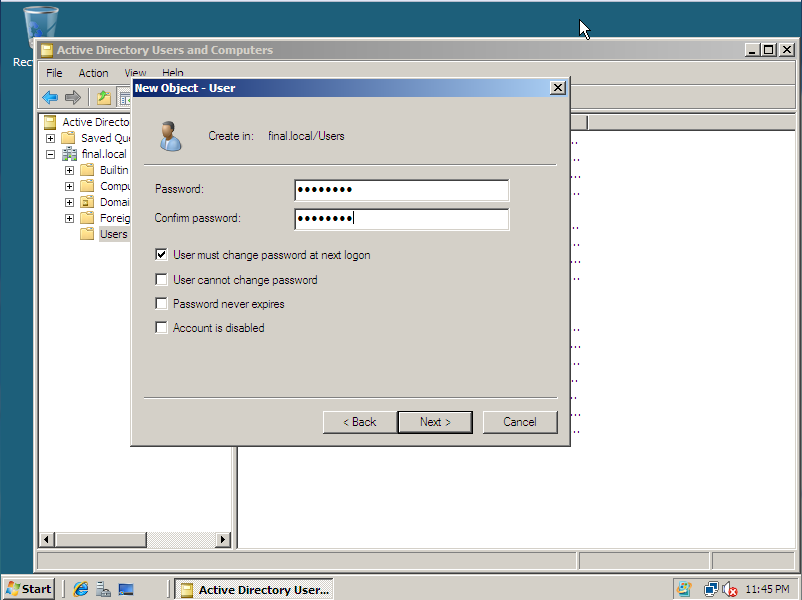
Figure

1. In the New Object window Enter the user’s information and click Next



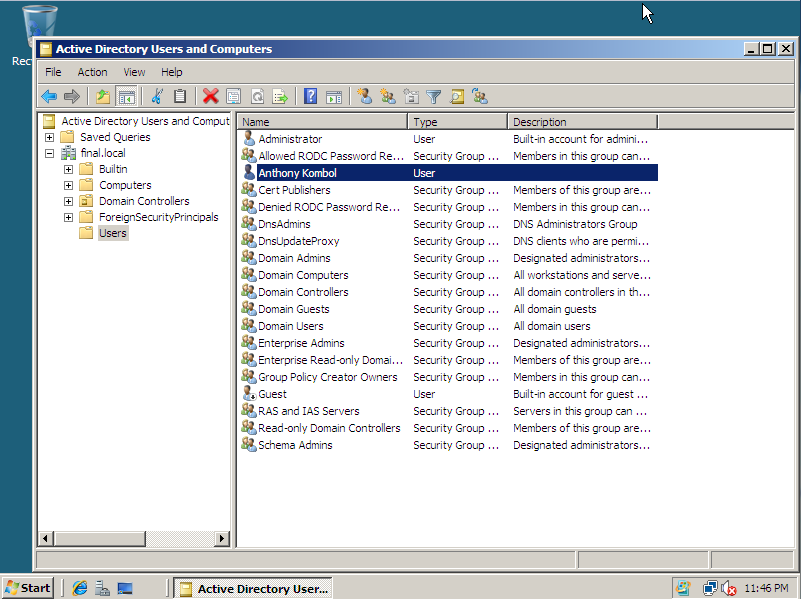
Figure

1. Enter an initial password for the user and make sure that the checkbox for “User must change password at next logon” is selected and click Next.



Figure

1. Finally click Finish, and you will see the new user in the right pane of the window.

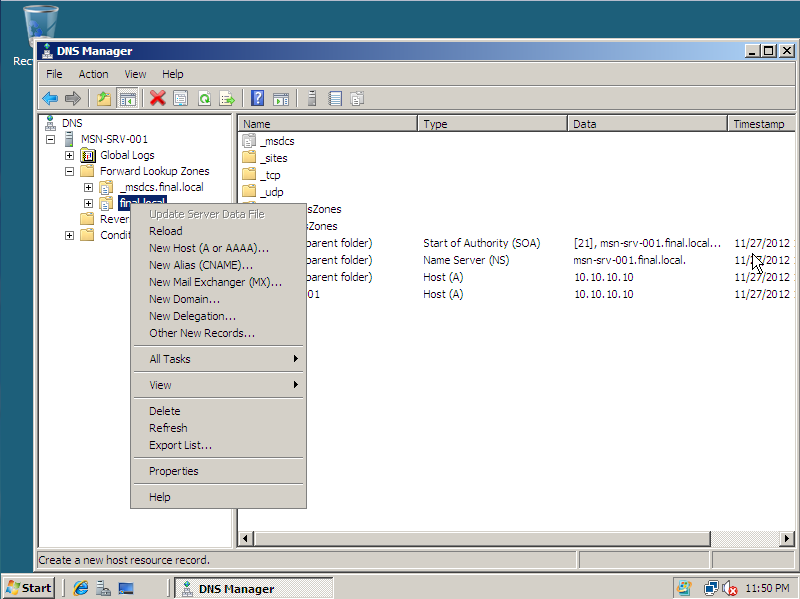
****

Figure

### Creating an A Record

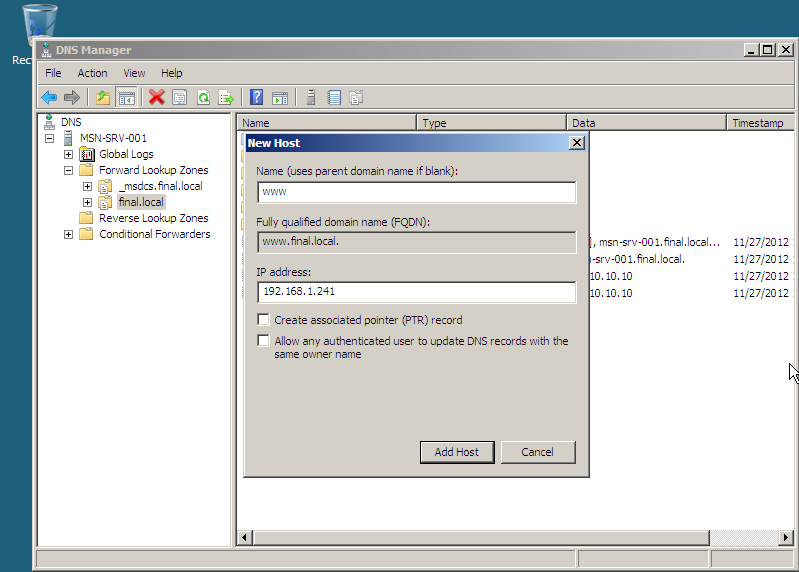
The following steps show how to create an A Record in Windows Server 2008

1. Open the Start Menu and select Administrative Tools > DNS
2. Expand the Server Name and Forward Lookup Zones and locate your domains forward lookup zone
3. Right click your forward lookup Zone andselect “New Host(A or AAAA)…”



Figure

1. Enter the host name for the device you are creating the record for in the Name field, and then enter the device’s IP address in the IP address field. Click Add Host.



Figure

1. Click OK at the success prompt and then click Done.
2. You can now see the new A record added in the right pane of the window.

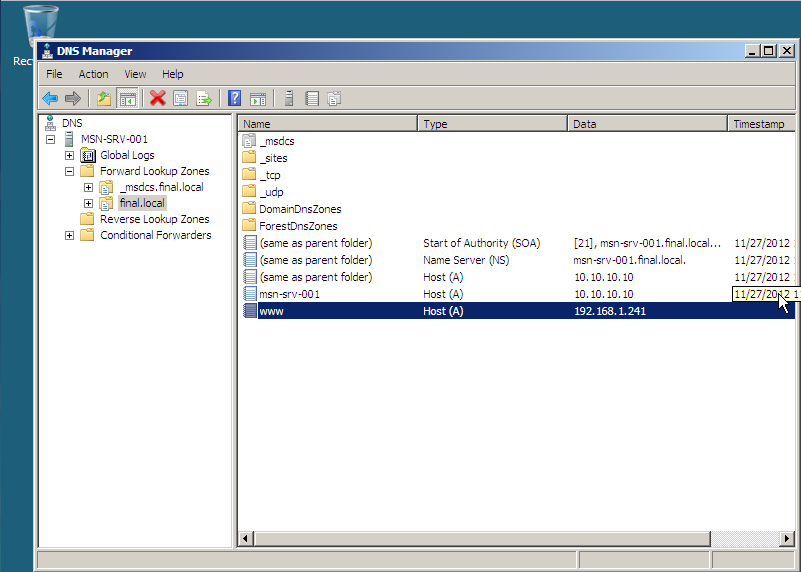


Figure A-1

## Database Server Setup

**Install MySQL**

## Web Server Setup

### Installing Apache and PHP

The process for installing Apache is rather trivial, however to include PHP in the mix we need a couple of extra packages installed. This one liner is as follows:

Table

|  |
| --- |
| **sudo apt-get install apache2 php5 libapache2-mod-php5** |

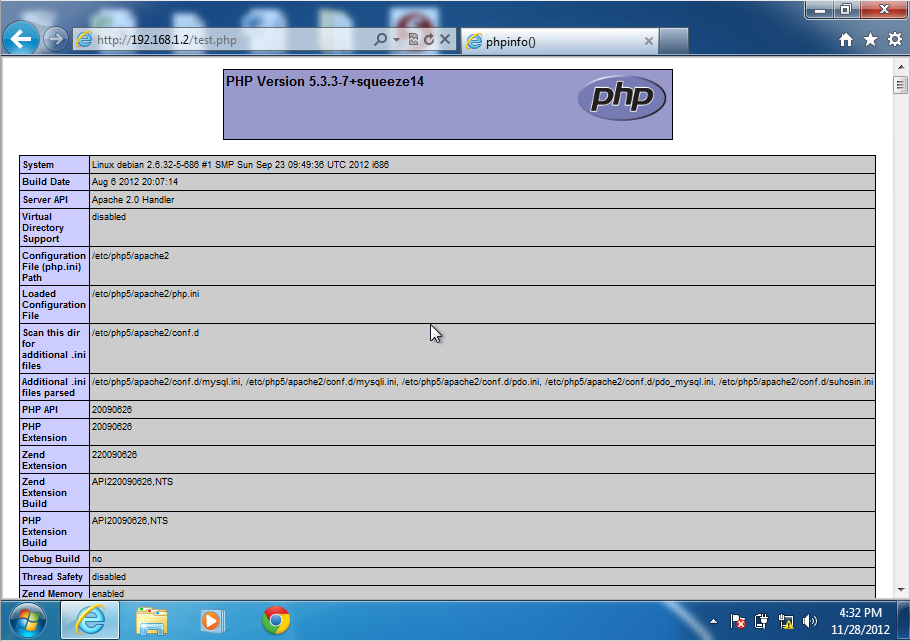
After this command is complete you should be able to open a web browser and go to the server’s IP address to see the default Apache page.

You may want to test to see if PHP is working correctly. To do this create a test.php page in /var/www and enter the following code:

Table

|  |
| --- |
| <?php  Phpinfo();  ?> |

After saving the file and going to http://<serverip>/test.php in your browser, you should see a page similar to this:

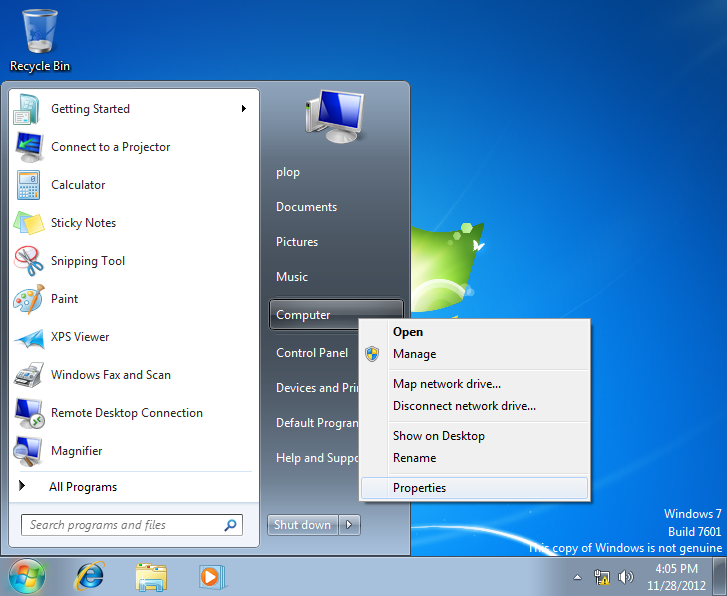


Figure

## Connecting Windows 7 Client to domain

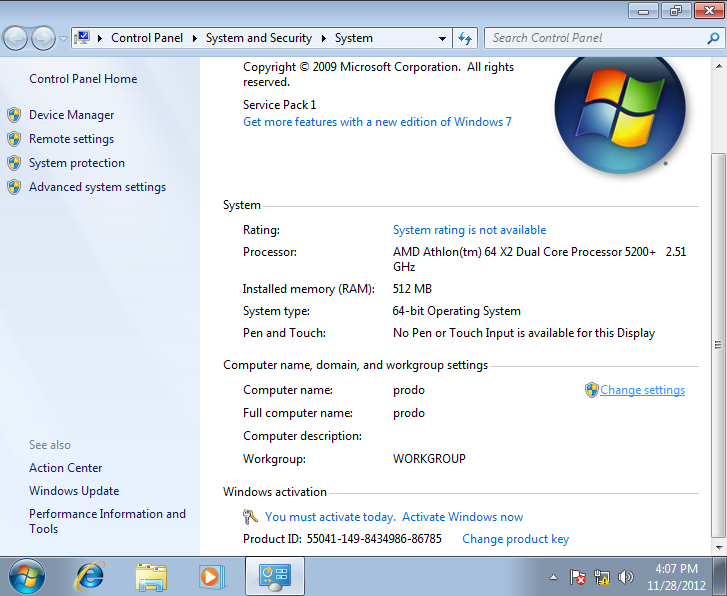
The following steps detail adding a client to the new Windows Domain

1. Open the Start Menu, right Computer and select Properties



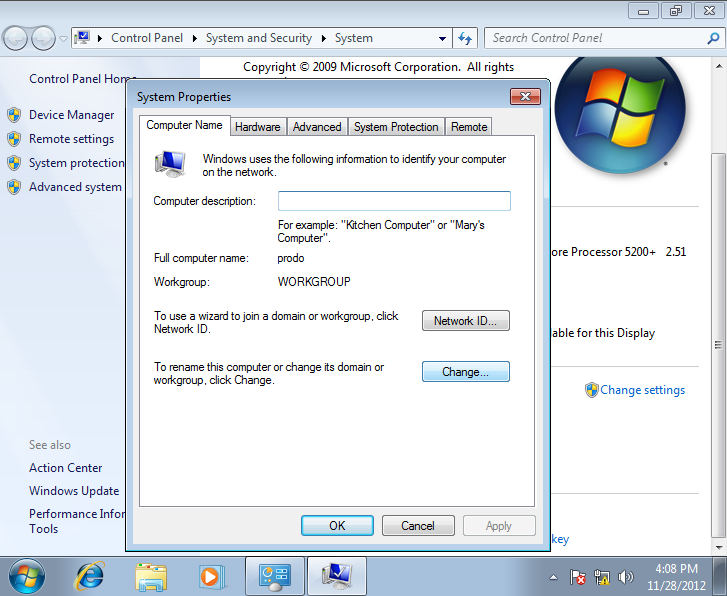
Figure

1. I the properties window, under “Computer name, domain and workgroup settings,” click the “Change Settings” link



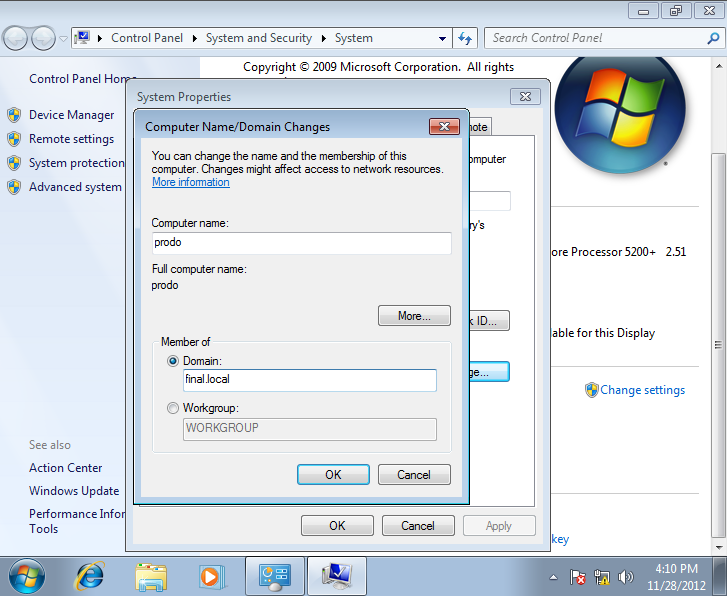
Figure

1. In the System Properties window, click the “Change…” button.



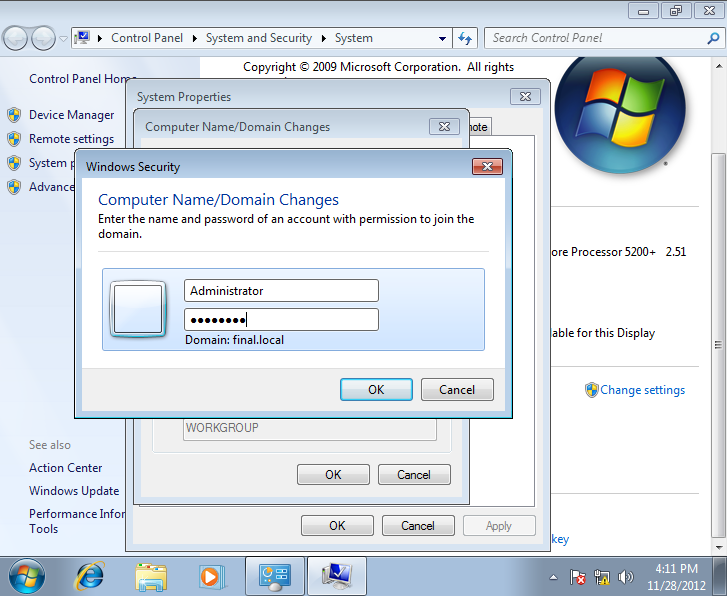
Figure

1. In the Computer Name/Domain Changes window, select the “Domain” radio button and type the domain name you are connecting to.



Figure

1. You will be prompted to enter the Domain Administrator’s username and password.



Figure

1. After entering the information, a welcome dialog box will appear. Click OK, you will be prompted to restart the computer. Click OK again, then Close, then Click Restart Now. The computer will restart and upon reboot, will be added to the domain. Any domain users will be able to login.

## Cisco 1841 Router Configuration

### Base Configuration

The first step to configuring our router was setting the host name of the device and password protecting it. The following command were used to do this. Comments are preceded with ! and explain what each command is doing. For simplicity of the project we used the password P@ssw0rd.

Table

|  |
| --- |
| !Configure hostname  hostname GateKeeper  !setup enable secret, console password, and telnet (vty) password  enable secret P@ssw0rd  line console 0  password P@ssw0rd  login  line vty 0 4  password P@ssw0rd  login  !Turn password encryption service on  Service password encryption |

Next, we configured the interfaces on the router…

Table

|  |
| --- |
| interface fa0/0  description EMPLOYEES subnet  ip address 192.168.1.30 255.255.255.224  no shutdown  interface fa0/1  description IT subnet  ip address 192.168.1.62 255.255.255.224  no shutdown  ! Set interface fa0/1/0 to full duplex to prepare it for sub interfaces  ! and dot1q encapsulation  interface fa0/1/0  description Trunk to DMZ/SecureZone  duplex full  no shutdown  ! Create sub interfaces and assign VLAN and ip addressing  interface fa0/1/0.2  description SecureZone subnet  encapsulation dot1q 2  !The command above encapsulates traffic for VLAN 2  ip address 192.168.1.230 255.255.255.248    interface fa0/1/0.3  description DMZ subnet  encapsulation dopt1q 3  ! The command above encapsulates traffic for VLAN 3  ip address 192.168.1.246 255.255.255.248  !configure interface for ISP link  interface fa0/1/1  description Link to ISP  ip address 174.110.255.254 255.255.255.0  no shutdown |

The final part of our base configuration is to create a default route

Table

|  |
| --- |
| ip route 0.0.0.0 0.0.0.0 174.110.255.253 |

This basically says that for any traffic the router receives, whose destination network has no known router, send it to 174.110.255.253, which is the ISP router interface.

### DHCP

The following configuration was used to setup DHCP services on the router. Again, a ‘!’ denotes a comment explain the command.

Table

|  |
| --- |
| !Create DHCP pool for Employees subnet  ip dhcp pool EMP  !Specify network range for DHCP pool  network 192.168.1.0 255.255.255.224!Specify default gateway to be assigned  default-router 192.168.1.30!Specify DNS server to be assigned  dns-server 192.168.1.225!Specify lease time  lease 1 0 0  !Specify reserved address not to be assigned by DHCP. These are static  ip dhcp excluded-address 192.168.1.26 192.168.1.30  !Create DHCP pool for IT subnetip dhcp pool ITnetwork 192.168.1.32 255.255.255.224default-router 192.168.1.62dns-server 192.168.1.225lease 1 0 0ip dhcp excluded-address 192.168.1.58 192.168.1.62 |

Since all our servers will have static address, we did not create DHCP pools for the DMZ or the SecureZone subnets.

**NAT**

The following commands were used to configure NAT on our router. We configured a one-to-many NAT for our department subnets and a one-to-one static NAT for our Apache web server.

Table

|  |
| --- |
| !Create NAT pool for department subnets  ip nat pool LOCAL 174.110.255.50 174.110.255.50 netmask 255.255.255.0 |

First we needed to create access lists that will be used to allow specific IP ranges to be NAT’d. Since we are NATing both department subnets, we used a supernet Wildcard mask to specify the IP range for both subnets. This was possible because those subnet ranges were side by side.

Table

|  |
| --- |
| access-list 1 permit 192.168.1.0 0.0.0.63 |

Next, we specify the access list for our inside addresses to be NAT’d and which interfaces are our inside networks and which interface is outside, or where the NAT translation will occur.

Table

|  |
| --- |
| ip nat inside source list 1 pool LOCAL overload  int fa0/0ip nat insideint fa0/1ip nat insideint fa0/1/1 ip nat ouside |

Below are the commands for the one-to-one static NAT for our web server

Table

|  |
| --- |
| ip nat pool WWW 174.110.255.51 174.110.255.51 netmask 255.255.255.0!We used the 0.0.0.0 wildcard mask to specify just one inside address  access-list 2 permit 192.168.1.241 0.0.0.0ip nat inside source list 2 pool WWWint fa0/1/0.3 ip nat insideint fa0/1/1 ip nat outside!This specifies that traffic sent to 174.110.255.51 on port 80 or 443 should  !be sent to the internal address 192.168.1.241 on those same ports  ip nat inside source static tcp 192.168.1.241 80 174.110.255.51 80ip nat inside source static tcp 192.168.1.241 443 174.110.255.51 443 |

### Access Control Lists

The following are our ACL settings with comments specifiying what they do.

Table

|  |
| --- |
| !Creates a named ACL for Internet traffic coming in  ip access-list extended UNTRUSTED !The following deny statement are to block traffic coming in from  !reserved addresses which signifies spoofing the source IP address  deny ip 10.0.0.0 0.255.255.255 any deny ip 172.16.0.0 0.15.255.255 any deny ip 192.168.0.0 0.0.255.255 any deny ip 127.0.0.0 0.255.255.255 any deny ip 169.254.0.0 0.0.255.255 any deny ip 100.64.0.0 0.63.255.255 any deny ip 192.0.2.0 0.0.0.255 any deny ip 198.18.0.0 0.1.255.255 any deny ip 198.51.100.0 0.0.0.255 any deny ip 203.0.113.0 0.0.0.255 any deny ip 255.255.255.255 0.0.0.0 any !This line blocks traffic from the outside destined to an internal  !IP address  deny ip any 192.168.1.0 0.0.0.255 !This allows pinging to our public IP addresses and allows  !internal traffic to ping outside  permit icmp any 174.110.255.50 0.0.0.1  !Allows external hosts to connect to our web server permit tcp any host 174.110.255.51 eq 80 permit tcp any host 174.110.255.51 eq 443 !Allows our internal clients to access outside web servers  permit tcp any eq 80 host 174.110.255.50 established permit tcp any eq 443 host 174.110.255.50 established  !Named ACL to control traffic coming out of the DMZ  ip access-list extended DMZ permit icmp any any  !allow the web server to talk to the MySQL server on the default port 3306 permit tcp host 192.168.1.241 host 192.168.1.226 eq 3306 !allow the web server to respond only to HTTP/HTTPS requests it receives  permit tcp host 192.168.1.241 eq 80 any established permit tcp host 192.168.1.241 eq 443 any established !Named ACL to control traffic coming out of the SecureZone subnet  ip access-list extended SECZONE permit icmp any any !Allow the MySQL server to respond only to the HTTP server  permit tcp host 192.168.1.226 eq 3306 host 192.168.1.241 !Allow the Windows Server to respond to DNS requests  permit udp host 192.168.1.225 eq 53 any !Allow Active Directory Authentication from department subnets  permit ip host 192.168.1.225 192.168.1.0 0.0.0.63 |

## Cisco Catalyst 2950 Switch Configuration

The following commands show our switch configuration. We started by configuring the hostname and access passwords for the switch.

Table

|  |
| --- |
| hostname S1  enable secret P@ssw0rd  line console 0loginpassword P@ssw0rdline vty 0 4loginpassword P@ssw0rdservice password-encryption |

Next we created VLANs on the switch

Table

|  |
| --- |
| vlan 2  name SecureZonevlan 3name DMZ |

Then we configured which interfaces would belong to each VLAN. We assigned interfaces 1-8 to VLAN 2 and 9-16 to VLAN 3

Table

|  |
| --- |
| vlan 2  name SecureZonevlan 3name DMZ  interface range fa0/1 - 8switchport mode accessswitchport access vlan 2!interface range fa0/9 - 16switchport mode accessswitchport access vlan 3 |

Finally, we needed to create a trunk port, which we assigned to interface 24 on our switch

Table

|  |
| --- |
| int fa0/24  !sets port to full duplex  duplex full  !assigned VLANs to trunk through this portswitchport mode trunkswitchport trunk allowed vlan 2,3 |