I decided to incorporate the Cisco 2600 into my previously designed network. This would give me two separate broadcast domains for future additions to the network, as well as a NAT point to prevent interfering with other machines on the school network due to IP address conflicts. I had already configured a Server 2008 box with AD DS and NPS, as well as a Firebox X Edge firewall. Thus the topology would look like this:
Basic Configuration of Cisco 2600 Router

The first step is to clear out the configuration of the Cisco 2600. I used a Serial connection to the console port of the 2600 with putty:
Basic Configuration of Cisco 2600 Router

Typing the following commands will reset the cisco to its default settings.

```
Router# erase nvram
```

This will erase the configuration stored in flash memory.

```
Router# erase nvram
```

Erasing the nvram filesystem will remove all configuration files! Continue? [confirm] [OK]

```
 erase nvram: complete
```

The router will then restart with the default settings. I recommend unplugging all networking cables at this point until you are done with the basic configuration.

You'll see information about the router's built in hardware, such as interfaces and memory. Eventually you'll see this:

```
--- System Configuration Dialog ---
Would you like to enter the initial configuration dialog? [yes/no]: n
```

Generally, this is not used, as it is very basic.
When you see this, you are ready to begin:

```
Router con0 is now available

Press RETURN to get started.
```

The first step is to enter Privileged EXEC mode, essentially the administrative mode:

```
Router> enable
```

Using the exit or end command will go back to the previous configuration prompt, when you exit configuration mode, it writes the changes:

```
Router(config)#exit
Router#*
```

I set the clock, which is important for logging functions and other things:

```
Router#clock set 10:45:30 7 aug 2013
```
Basic Configuration of Cisco 2600 Router

Now for the fun stuff: Configuring and enabling the interfaces:

```
Router#show ip interface
FastEthernet0/0 is administratively down, line protocol is down
   Internet protocol processing disabled
FastEthernet0/1 is administratively down, line protocol is down
   Internet protocol processing disabled
Ethernet1/0 is administratively down, line protocol is down
   Internet protocol processing disabled
Router#
```

As you can see, all interfaces are down and they are currently disconnected physically as well. You need these names to configure the interfaces though. Enter config mode:

```
Router(config)#interface fastethernet0/1
Router(config-if)#ip address 10.16.79.1 255.0.0.0
Router(config-if)#end
Router(config)#interface ethernet1/0
Router(config-if)#ip address dhcp
Router(config-if)#end
Router(config)#
```

Let's also configure the default gateway used by the router:

```
Router(config)#ip default-gateway 172.16.0.1
```

And the default name server, which would likely be an internal DNS but google for this example:

```
Router(config)#ip name-server 8.8.8.8 8.8.4.4
Router(config)#exit
```

Aug 7 11:58:23.669: %SYS-5-CONFIG_I: Configured from console by console

Aug 7 11:54:53.248: %SYS-5-CONFIG_I: Configured from console by console
I began to notice these following errors:

```
Router#config
Configuring from terminal, memory, or network [terminal]?
Enter configuration commands, one per line. End with Ctrl/Z.
Router(config)#interface ethernet0
result (config)#duplex auto
result (config)#duplex full
result (config)#exit
result (config)#exit
result#
```

We can fix this problem by using this command from the externals interface prompt:
Let's confirm our settings. Use the `show config` command from the EXEC prompt. Please note that I added this after completing the experiment, and so some info should not be there, ignore the settings I have not highlighted specifically:
Now that both interfaces are up and the gateway and DNS are configured, I connected the cat5 cables to the interfaces. I pinged the internal firewall: 10.16.70.2, and the external firewall: 172.16.0.1.

```
Router#ping 10.16.70.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 10.16.70.2, timeout is 2 seconds:
 !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms

Router#ping 172.16.0.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.0.1, timeout is 2 seconds:
 !!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/1/4 ms
```
Basic Configuration of Cisco 2600 Router

Now we know we have connectivity. Let's test from the server:

```
PS C:\Users\Administrator> ping 192.168.0.1
Pinging 192.168.0.1 with 32 bytes of data:
Reply from 192.168.0.1: bytes=32 time=78 TTL=64
Reply from 192.168.0.1: bytes=32 time=78 TTL=64
Reply from 192.168.0.1: bytes=32 time=78 TTL=64
Reply from 192.168.0.1: bytes=32 time=78 TTL=64
```

Ping statistics for 192.168.0.1:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 78, Maximum = 78, Average = 78

```
Pinging 192.168.0.1 with 32 bytes of data:
Reply from 10.18.70.11: bytes=32 time=2ms TTL=254
Reply from 10.18.70.11: bytes=32 time=2ms TTL=254
Reply from 10.18.70.11: bytes=32 time=2ms TTL=254
Reply from 10.18.70.11: bytes=32 time=2ms TTL=254
```

Ping statistics for 10.18.70.11:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 2, Maximum = 2, Average = 2

```
Pinging 172.16.66.18 with 32 bytes of data:
Reply from 172.16.66.18: bytes=32 time=2ms TTL=254
Reply from 172.16.66.18: bytes=32 time=2ms TTL=254
Reply from 172.16.66.18: bytes=32 time=2ms TTL=254
Reply from 172.16.66.18: bytes=32 time=2ms TTL=254
```

Ping statistics for 172.16.66.18:
Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
Minimum = 2, Maximum = 2, Average = 2

```
Pinging 172.16.8.1 with 32 bytes of data:
Request timed out.
```

Ping statistics for 172.16.8.1:
Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
So we can see that the server cannot ping the last firewall, but we saw previously that the Cisco router can. This is why we need to set up NAT. Let's go back to the interface configs for the two interfaces we are using:

```
router(config)#interface ethernet 0/0
router(config-if)#ip nat outside
```

This takes a bit of explaining. NAT allows us to automatically translate IP addresses based on certain criteria, which could be specific ports, addresses, etc. I chose a simple static NAT, which in this case will simply route all traffic received internally to the external interface using the external IP. The IP address I used is the external of the firewall, which means all traffic coming from the firewall is routed out of the external interface of the Cisco as shown.

This NAT works for this simple experiment, but would require a more complex configuration in most real-world deployments. There is a catch to this setup I will show you soon.

First, let's verify all of our configuration settings with the `show ip NAT translation` command:
Basic Configuration of Cisco 2600 Router

It is configured properly, so let's test again with our server by pinging the external firewall:

```
Pinging 172.16.0.1 with 32 bytes of data:
Reply from 172.16.0.1: bytes=32 time=1ms TTL=62
Reply from 172.16.0.1: bytes=32 time=1ms TTL=62
Reply from 172.16.0.1: bytes=32 time=1ms TTL=62
Reply from 172.16.0.1: bytes=32 time=1ms TTL=62

Ping statistics for 172.16.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss).
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 1ms, Average = 1ms
Pinging 172.16.0.1 with 32 bytes of data:
Reply from 172.16.0.1: bytes=32 time=1ms TTL=62
Reply from 172.16.0.1: bytes=32 time=1ms TTL=62
Reply from 172.16.0.1: bytes=32 time=1ms TTL=62
Reply from 172.16.0.1: bytes=32 time=1ms TTL=62

Ping statistics for 172.16.0.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss).
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 1ms, Average = 1ms
PS C:\Users\Administrator> ping www.google.com
Pinging www.google.com [74.125.137.147] with 32 bytes of data:
Reply from 74.125.137.147: bytes=32 time=108ms TTL=38
Reply from 74.125.137.147: bytes=32 time=122ms TTL=38
Reply from 74.125.137.147: bytes=32 time=118ms TTL=38
Reply from 74.125.137.147: bytes=32 time=108ms TTL=38

Ping statistics for 74.125.137.147:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss).
    Approximate round trip times in milli-seconds:
        Minimum = 108ms, Maximum = 122ms, Average = 117ms
PS C:\Users\Administrator> ping 172.16.0.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.0.1, timeout is 2 seconds:
....
Success rate is 0 percent (0/5)
Router#
```

Let's confirm we now have web access:

```
PS C:\Users\Administrator> ping www.google.com
Pinging www.google.com [74.125.137.147] with 32 bytes of data:
Reply from 74.125.137.147: bytes=32 time=108ms TTL=38
Reply from 74.125.137.147: bytes=32 time=122ms TTL=38
Reply from 74.125.137.147: bytes=32 time=118ms TTL=38
Reply from 74.125.137.147: bytes=32 time=108ms TTL=38

Ping statistics for 74.125.137.147:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss).
    Approximate round trip times in milli-seconds:
        Minimum = 108ms, Maximum = 122ms, Average = 117ms
PS C:\Users\Administrator>
```

And for the catch, and why a more advanced configuration would be necessary outside of my test network:

```
Router#ping 172.16.0.1
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 172.16.0.1, timeout is 2 seconds:
....
Success rate is 0 percent (0/5)
Router#
```

The Cisco can no longer ping the firewall, because the return pings are routed out of the internal interface, however this is acceptable for this test network. As I expand I may change this to a dynamic NAT that only NATs web traffic.