The student, the professor, and Optix Pro

GCIH Practical Assignment v3.0

By: Don Parker, GCIA

SANS Parliament Hill August 6-11, 2004

Table of Contents

Part 1

Statement of Purpose

The contents of this paper are being submitted to fulfill the practical portion of the GCIH certification process. This paper is broken down into two main parts which consist of the “exploit in action” and the “incident handling” process.

There have been many exploits researched and analyzed by previous GCIH certified analysts. Of the papers on the GIAC website for the GCIH many are based on buffer overflows, and other similar type exploits. In essence operating system flaws whether they be in the operating system itself, or a flaw in a companion program such as an ftp server.

What this paper is based on however is not a system or program flaw, but rather on that domain of system compromises known as the trojan. Where these programs used to be crude in nature they have now become far more sophisticated, and lethal. The popularization of the trojan per my perspective began with the well known, and very well designed Back Orifice back in 1998. It was written by one of the most talented group of hackers in its day who were known as the Cult of the Dead Cow.

Things have changed though since the release of Back Orifice, and the many successors to it. The security professional now has to worry about far more dangerous variants of the original design concept. Though these Trojans or RATS are widely used by what are termed by the hacking community as “script kiddies” they are most certainly not written by them.

This brings me back to my reason for choosing a modern and very dangerous trojan which is known as Optix Pro v1.32 I wanted to look at a piece of malware which posed a real threat to not only home users, but small company offices as well. That being said I don’t really see any type of trojan lasting long in a true corporate environment. Typically a corporate LAN is well protected on not only the interior, but also the interior.

What makes this trojan dangerous you ask? The simple reason is that it contains not only firewall and anti-virus killing capabilities. It also has the ability to detect and disable trojan removal programs. Lastly it also gives the person the ability to kill specific executables as well. One example of an executable someone of malicious intent may want to kill as well is say the program fport or activeports.

Another objective of this paper is to show just how easy it is to use one of these trojans. While I will not be doing any binding of a trojan to a small executable for infection I will however show another infection vector. One which is largely ignored and relegated to the sidelines. That of physical access.

It is the intent of this paper to show just how important physical security is as well. Most people are concerned with hardening the exterior of the lan with a router followed by firewalls and IDS systems. Quite often little attention is paid to actually physically securing the computer itself from unauthorized access. Shown over the course of this paper is the danger involved in ignoring this often neglected aspect of computer security; physical access. The scenario that I use in this paper is that of a student infecting his professor's computer with a trojan. This student does so that so that he can gain access to the upcoming exam, which he believes is stored on the professors computer.

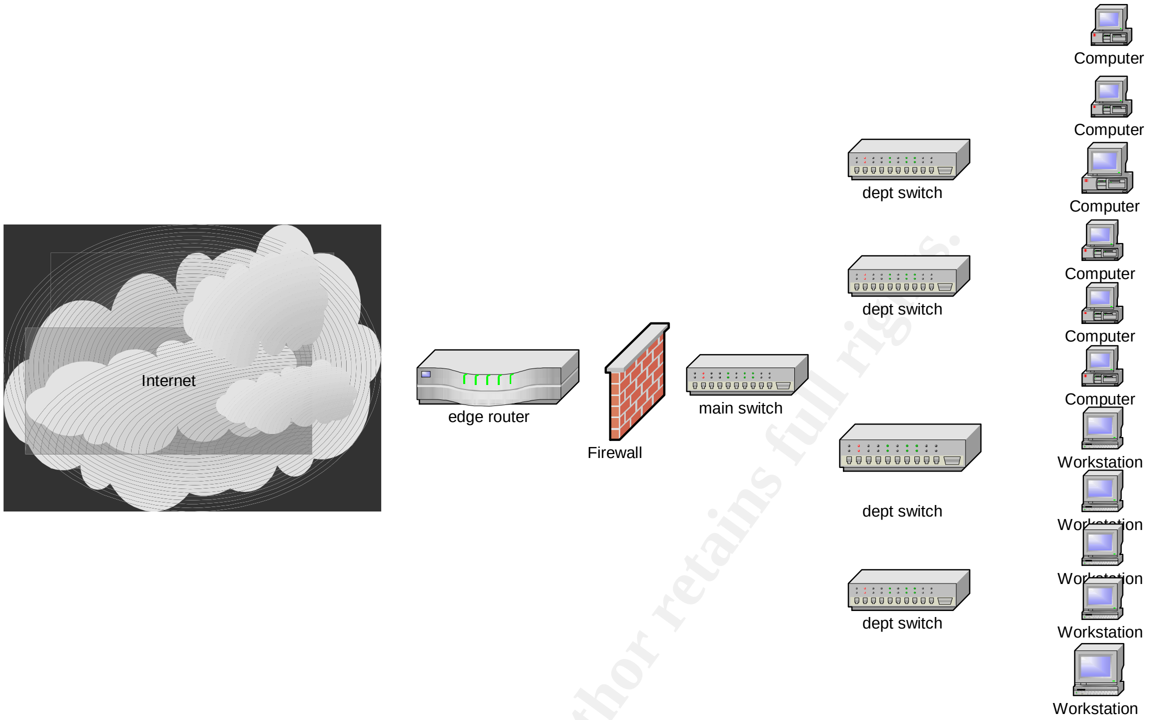
In this case the student realizes that he can have access to the professor's computer via the class he attends. He also realizes that the professor also has an open door policy for his office. Bearing all of this in mind the student has chosen to physically infect the computer by plugging in his USB stick, and copying the trojan server onto the professor's computer. He will then be able to control and manipulate the computer while he is in class being taught by the professor himself. Quite clever the student thinks as the professor will not notice any odd behavior on his computer as he will be teaching class. Problem being of course is that like many who use these tools they are not as clever as they think they are. We will see over the course of the paper just how and why the student is caught.

Next on the agenda is a view of the topology, which will be used for the simulated trojan scenario. This is a crucial part of the exploit itself working, as the student realizes that he can exploit the professor's computer due to the university network setup. I alluded earlier that the student can access the professor's computer when he is in the classroom. This is done via CAT 5 drops the university has provided. These CAT 5 drops are wired into the desks located in the classroom. This was done as a courtesy on the universities behalf as they thought it quite elegant that the students could communicate with the professor's computer in that way, and allowing them file sharing access. Problem as well is that the students would also be able to access any other listening sockets on the professor's computer.

There is a specific reason I chose the university, or college as a backdrop. It is widely acknowledged that these academic institutions have a very delicate balancing act to do. Providing network access to their students is a must, but they must also try and balance security with that access.

It has been my experience that this is still very much an ongoing learning process for these academic venues. Trying to achieve the right balance of user freedom while striving for network security is a very difficult juggling act. This is in light of the fact that many of these academic institutions are circumvented from within. Such endemic problems as P2P file sharing, and lack of anti-virus & firewalls on student computers almost always invariably leads to an incident.

Bearing all of the above in mind is why I chose to simulate a university campus network setup. Please see the below noted diagram for a visual representation followed by an explanation of the network itself below the diagram.



As seen above we have the internet, as shown by the cloud. Then there is the universities edge router, or first entry point if you will. Behind the edge router is a firewall, which in turn leads to the main switch. Hanging off of the main switch are the various departmental switches. Though they are only four listed here there in reality there would be one for each department such as History, Science, and so on.

Each of the departmental switches is wired to that department's main amphitheater. At each of the desks in the amphitheater is a CAT 5 jack that the students can plug their laptops into so that they can access assignments off the instructor's computer. As mentioned above each of these CAT 5 jacks in the amphitheater all lead to the departmental switch.

Having said that, all of the file-sharing ports are indeed open, but this vector is not the ingress point for the exploit chronicled in this paper. In essence once the student has plugged into the available port in the classroom they have access to the professor's computer. Obviously this is not a desirable environment, but there really are such setups out there in the academic world today.

<u>NOTE</u>

On the edge router DHCP is enabled for assigning of IP addresses. Please note however that on the switches themselves some of the IP addresses are static ie: they do not change. There still is though DCHP enabled on the remained of non-mapped IP addresses. The static IP addresses are in use in all of the college's classrooms. Each desk has a static IP which does not change. At the beginning of the school year each of the students were made aware of what IP address went with what desk. These addresses flowed in a logical sequence in case they were to sit elsewhere in the room.

Lastly each of the students was at the beginning of the year given an image of Windows 2000 Professional courtesy of the college. The college has an enterprise license which allows for this. Part of the baseline image from the college also included an anti-virus program also provided as part of an enterprise license. Having this baseline helped simplify things for the college. It was also easier as well to show the students once on how to input a specific IP address upon entering a certain class. This was a somewhat inelegant solution, but one which the university was happy with.

Having a student assigned a specific IP address when sitting at a specific desk also will end up paying dividends later for them. Though they did not realize it this policy is what will catch them a cheating student.

Part 2

The Exploit

Name

The exploit used in this paper is actually a trojan, which goes by the name of Optix Pro v1.32. A trojan is not like a buffer overflow which will exploit an operating system weakness, or a virus that tries to replicate itself. A trojan by and large will just sit on the victims computer and open up a backdoor. It will almost always as well make some registry changes to the victims computer as well. These registry changes are so that the trojan itself restarts every time the computer is powered down, and restarted. There is no CVE number, CERT number, or Bugtraq release that was found, which could be associated with Optix Pro.

Optix Pro versions; there are a variety of them beginning with Optix Pro 10 and is currently up to v1.32 which is what I am working with. Once installed the trojan makes some registry entries;

HKEY_LOCAL_MACHINE\SOFTWARE\Microsof\windows\CurrentVersion\run\msiexec16.exe
HEKY_LOCAL_SOFTWARE\Microsoft\Windows\CurrentVersion\RunServices\msiexec16.exe

These registry entries allow Optix Pro to be restarted every time the computer itself is rebooted or powered back on.

The trojan will upon installation be installed to the C:\Winnt\System32 on a Windows 2000 computer. The name of the trojan server itself once built can be named to whatever the hacker chooses. Normally something innocuous is given to the Trojan server. Most every program in use today observes the client/server model for communications. This trojan is no different in that aspect as the infected machine will have the trojan server on it being controlled by the trojan client on the hackers computer.

Operating System

All of the following operating systems are vulnerable to Optix Pro regardless of service pack, or patch issued by Microsoft. Windows 95, Windows 98, Windows ME, Windows NT, Windows 2000, Windows XP. Windows 2003 is also said to be susceptible as evidenced by Sherman Hung's paper, which looked at Optix Pro v1.31. http://www.giac.org/practical/GCIH/Sherman_Hung_GCIH.pdf

Protocols/Services/Applications

The Optix Pro trojan uses TCP as its transport protocol for communication between both the victim host, and the controlling computer. There is nothing else remarkable about the trojan in and of itself, as it relates to this section. Mentioned earlier was the fact that a trojan does not operate like a buffer overflow, or format string attack. Due to this there is little else to add to this section in regards to Optix Pro itself. Though it should be noted that with a default "installation" of Optix Pro the computer will listen on TCP port 3410 for connections. This default port setting though can be changed. Further on in this paper are packet traces verifying the use of TCP as the transport protocol.

Variants

There are no known variants to Optix Pro itself. It should be noted though that there have been successive releases of this trojan ie: v1.3 v1.31 v1.32 and the such. Each successive release has incorporated fixes, or new features as it were. Seemingly bundling firewall/anti-virus killing capability into trojans nowadays is becoming more common. Optix Pro is not the only one with this capability as Beast also has it among others.

Description

To clarify again that the Optix Pro trojan being used is not an exploit per se, but rather could be considered as a legitimate remote administration tool. Due to this difference there is not a great deal of clarification or detail for this section. In reality the sole purpose of this trojan is to open a port. This trojan also has the built in ability to do much more such as file transfers and the such. If you will, think of Optix Pro much like VNC, which is also very much a legitimate remote administration tool. That being said VNC is often used by the malicious hacker to manipulate a victim's machine once they gained control of it. This is why a trojan like Optix Pro is so dangerous. Simply put because it has so many various functions built into it. Features such as an ftp server amongst other features.

Signatures of the Attack

With the default settings left in place this program is known to listen on TCP port 3410. This could be viewed as a signature for an IDS, but then again the trojan allows you to change the port that it will spawn a socket on when building the trojan server. As long as the attacker chooses an ephemeral port to listen on it is relatively easy to find. This is assuming that you are logging all packets on your network. Not everyone does this though due to the cost of having enough hard drive space to do this. Beyond this one possible signature of ephemeral port usage for the listening socket there is not much to give this trojan away at the packet level. (We will see later in the paper that a very effective IDS signature could be written to detect Optix Pro on an ascii content match)

Though once the trojan server has been installed on the victims computer it will by default install itself to one of a couple of places varying upon the operating system in use. This was detailed in that for Windows 2000 it will be install itself to C:\Winnt\System32. Please note as well that if someone is using a program such as activeports then they would notice as well the program msiexec16.exe show up in output of the program as well as in netstat if used (though only the port number for netstat usage). Please bear in mind a lot of this will be shown in the upcoming pages where I show the system being exploited.

The Platforms/Environments

Victims Platform

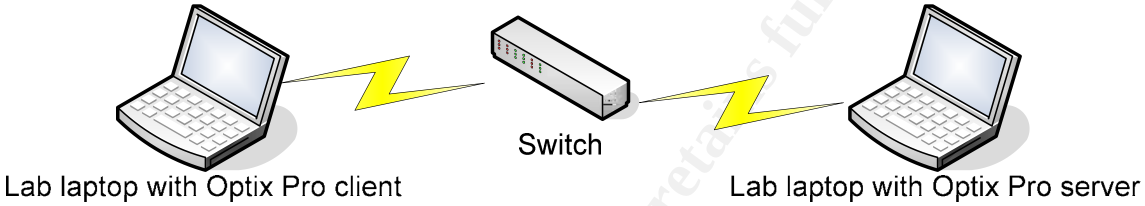
The operating system in use by the professor is Windows 2000 Professional with service pack 4 installed. A standard browser of Internet Explorer and mail client of Outlook Express is also in use. There is no firewall in place, but an anti-virus solution is installed on the victim's computer. This computer setup is very much the same baseline that the students are given for use by the college. Beyond this there is nothing else server wise installed on this computer such as an FTP server or other such service. Please note once again that this computer has all of the NetBIOS services ie: port 137-139 enabled. This was done so that the students in the classroom once plugged into the network could access the shares being offered by the professor's computer.

Source Network

I was uncertain at this point what was meant by source network after consulting the administrivia so I decided to be verbose and cover all of my bases. The source network where this exploit was practiced and played with was in my own home lab. Setup of the lab for this trojan based exploit was simply two laptops networked together via a switch/router ie: Linksys home switch/router.

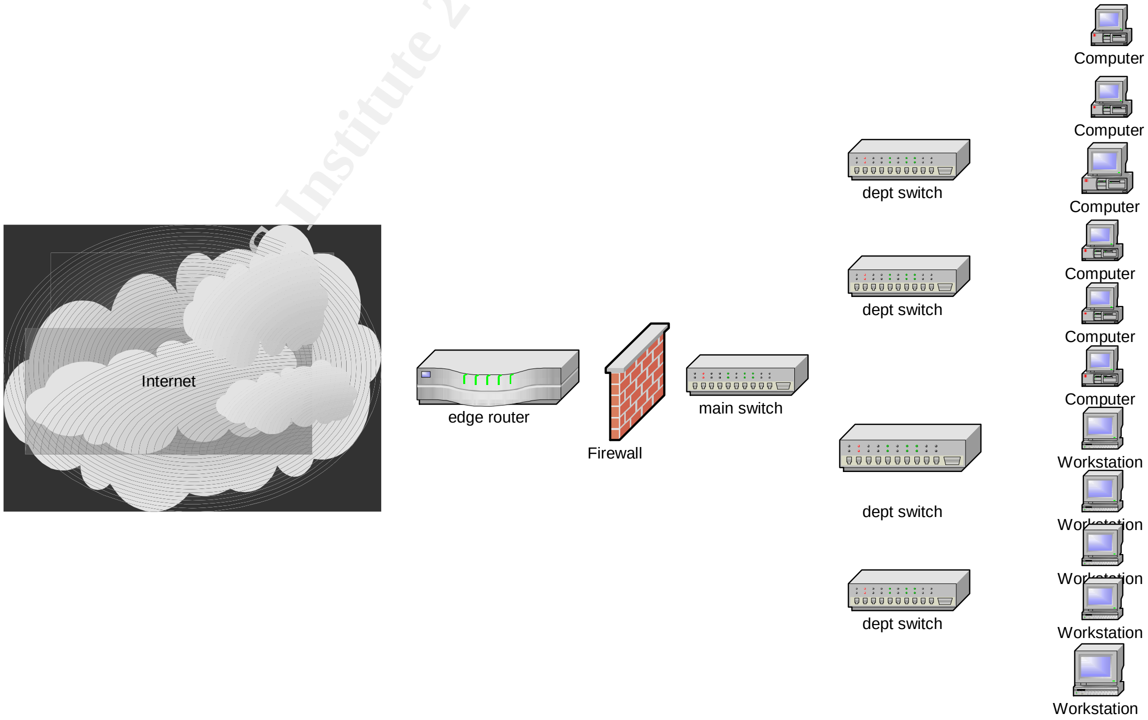
Nothing further was required as the packets generated by the trojan would have the proper TCP and IP headers built for it by the computers TCP/IP stack. In essence there was absolutely no need to actually try and infect then control a computer via the internet.

As seen below is how the lab setup was done;


Switch
Lab laptop with Optix Pro client
Lab laptop with Optix Pro server

The Optix Pro client was the one controlling the laptop infected with the Optix Pro server. Please note as well that both laptops are running Windows 2000 Professional.

Target Network



This as discussed earlier in the paper is pretty much a standard college or university topology. In actuality the local college in my hometown has much the same layout. As noted above it is a relatively flat network. All IP addresses are obtained via the router which itself employs DHCP to do assign IP addresses as required. However please note as mentioned earlier though that the classrooms themselves have static IP addresses.

Behind the edge router is the firewall itself which in reality is an application layer firewall. Behind the firewall itself is the main network switch. This switch in turn has the various departmental switches plugging into it.

In each of the various academic departments there are the actual client work stations there. In the case of this paper all of the departments have their classrooms setup with a CAT 5 drop at each desk. This enables the students to access their respective professor's computer for homework assignments and such. Students at the beginning of the year are given a diagram showing the IP address assignment of each desk in the classroom. This helps them assign themselves the IP address of their respective desks.

So as we can now see this is a relatively simple network. The one thing not discussed here are the mail servers as they are not of any importance to this paper. One last thing of note is that the edge router itself has a pretty standard set of ACL's on it to restrict access at the outer periphery. Only services being offered by the college are open on the router. All other ports are closed by design for security.

Stages of the Attack

Reconnaissance

For the purposes of this paper we will be chronicling the efforts of a rather lazy student who would rather try and cheat then write an exam. This student however does have some smarts, but not enough to escape detection. The though processes of the fictional student are as follows;

John had an upcoming math exam he did not particularly feel like writing. He much preferred hanging out in the IRC channels and trojan board forums. With the time he had been spending there he believed he was now a pretty "clued up" hacker. John realized that when attending his classes he was able to plus into his CAT 5 port and be a part of the college's network. Not only that he realized but he was also able to see the professor's computer seen as it was setup via NetBIOS to have file sharing enabled.

Though John did not know very much about the actual topology of the entire network he knew that he could access the computer from the comfort of his classroom. Not only that but also while the professor himself was not at his computer. This led John to think that he might just infect the professor's computer with a trojan and then download the upcoming exam. How to do it though John thought? He decided he would have to do some form of reconnaissance. Bearing this in mind he knew his knowledge of the network was insufficient to compromise the school from home. Bypassing the router and firewall was just not an option for him due to his limited knowledge.

John also knew from having seen the professor's computer before that it was a standard computer ie: Intel 0x86 architecture and not a Sparc. This was key as his trojan would only work on an Intel platform with a version of Windows operating system. From having previously seen it he also knew that the professor was running Windows 2000 Professional as well.

Once again he was back to how to do the hack and compromise the professor's computer. As he sat in class listening to the professor drone on he realized that it was really all about Occam's Razor. If the professor was in class and not at his computer then maybe he could just physically infect it. Why try and cut through the hardened exterior of the network if he had physical access to the computer. This made far more sense to him as he knew the professor had an open door policy with his students. Literally an open door! The professor kept the door to his office open at all times during school hours.

With this in mind John got up and left the classroom and walked down the 15 or so feet to the professor's office. Lo and behold the office door was indeed open and no one was in sight! John now had his way into the network and more specifically the professor's computer. The only worry John had was if another professor or faculty staff stopped by the office while John was in it and the math professor himself was in class teaching.

Being caught red handed was not an option. John decided that if caught he would simply say he had excused himself from class as he was feeling ill and was leaving the professor a note on his computer for him. John thought this was quite clever.

Scanning

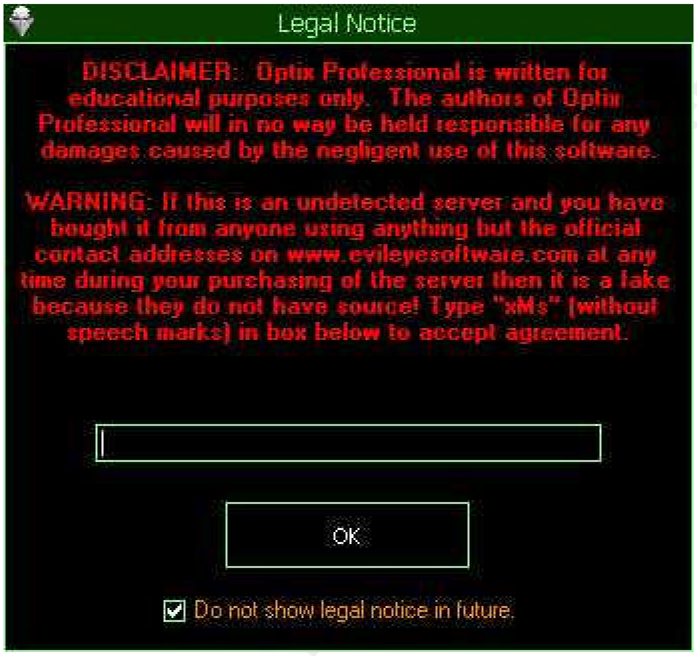
John returned to the class and now decided to get the professors IP address once he was plugged back into the network. Wait a minute! He remembered at the beginning of the semester the professor has handed out a network diagram showing the IP addressing scheme for the classroom and the professor's own IP address as well.. He already had it! This really was too easy John thought. He of course needed the IP address of the professor's computer so that he could control it once infected via the trojan client interface. Things were looking up John thought. It had been very easy so far to plan out his attack and get all the information necessary. John did not realize that these types of attacks were always easier when executed by someone who already had inside access to the network. None the less he thought himself quite clever.

Exploiting the System

John wanted to practice installing the trojan he was going to use several times first so he could do it as quickly as possible; therefore minimizing the risk of getting caught. He also wanted to configure as few options as required on the trojan server he was going to install on the professors computer. With that in mind he spent several hours practicing how to install the trojan server till he was comfortable and quick doing it.

John was now ready to go ahead and do the trojan installation. The next day he was in the class and let the professor go about 10 minutes into his class and then walked out. He quickly walked past the professors office to make sure no one was in. Then he quickly turned around and walked in. His first step was to simply disable the anti-virus on the professor's computer. This way his transferring of the trojan server onto the professors computer would not trigger any possible alarms just in case the anti-virus on the computer sent a message somewhere.

Quickly he inserted his USB key into the computer and copied the server onto the hard drive. Now he double clicked it and up came the window as seen below. Inputting the characters he then moved on to the actual configuration of the server itself.



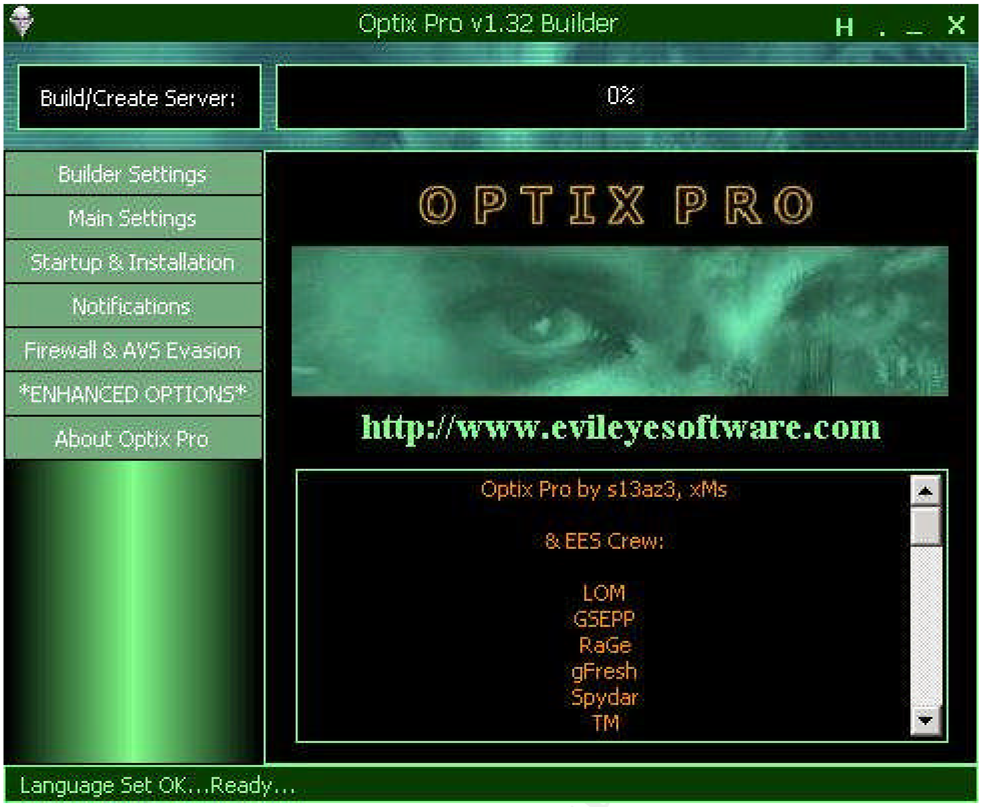
Legal Notice

DISCLAIMER: Optix Professional is written for educational purposes only. The authors of Optix Professional will in no way be held responsible for any damages caused by the negligent use of this software.

WARNING: If this is an undetected server and you have bought it from anyone using anything but the official contact addresses on www.evileyesoftware.com at any time during your purchasing of the server then it is a fake because they do not have source! Type "xMs" (without speech marks) in box below to accept agreement.

☑ Do not show legal notice in future.

Once he had entered the pass as noted above up came the next window



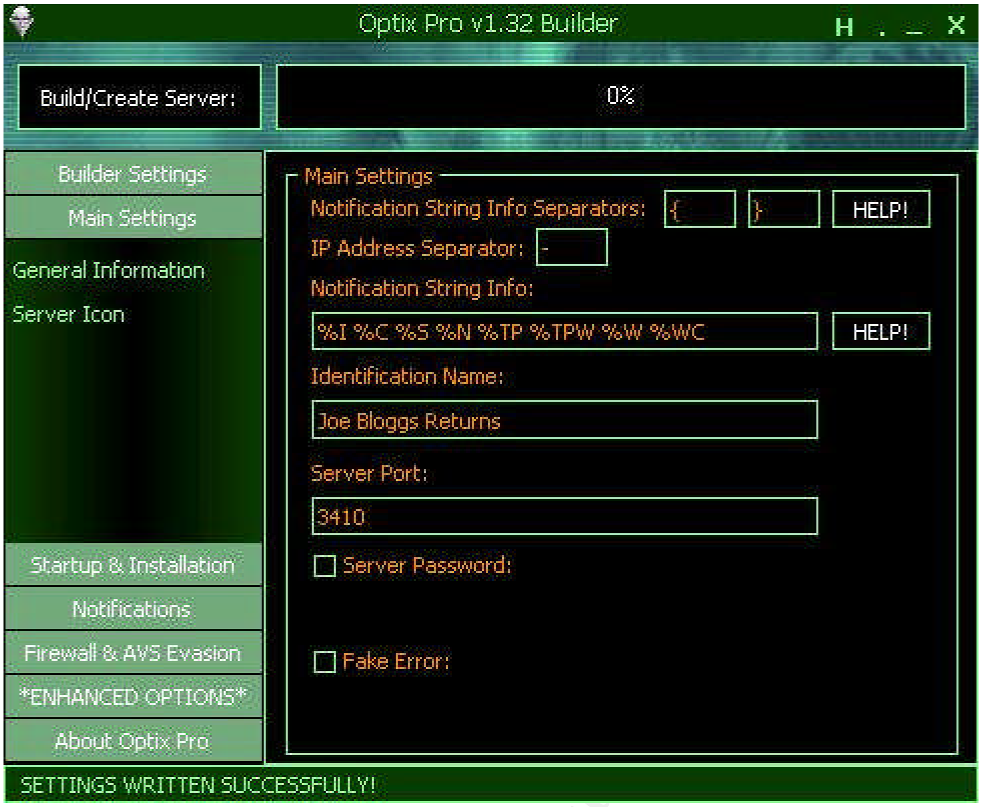
Optix Pro by s13az3, xMs

& EES Crew:

LOM
GSEPP
RaGe
gFresh
Spydar
TM

Language Set OK...Ready...

At this point John had to quickly go through the settings for the server itself. The main ones being under "Main Settings". In this sub menu he set the default port that Optix Pro server should listen on for connections, and also a server password which he did not bother with.



Main Settings

Notification String Info Separators: { }

IP Address Separator: -

Notification String Info:

%I %C %S %N %TP %TPW %W %WC

Identification Name:

Joe Bloggs Returns

Server Port:

3410

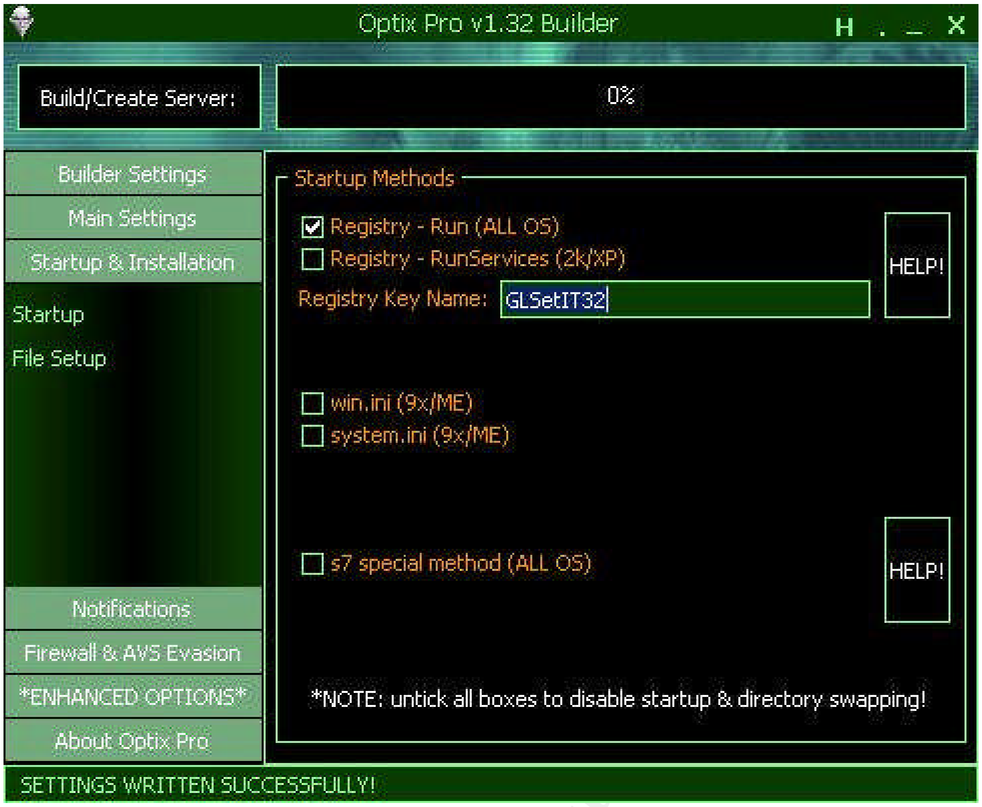
□ Server Password:

□ Fake Error:

SETTINGS WRITTEN SUCCESSFULLY!

Next he chose an icon with which the Optix Pro would be saved as. This would help the professor from stumbling across it and wondering what it was. There was a wide variety of icons to save the server as.

Once this was done he quickly went onto the “Startup & Installation” menu.



Startup Methods

☑ Registry - Run (ALL OS)
☐ Registry - RunServices (2k/XP)
Registry Key Name: GLSetIT32

☐ win.ini (9x/ME)
☐ system.ini (9x/ME)

☐ s7 special method (ALL OS)

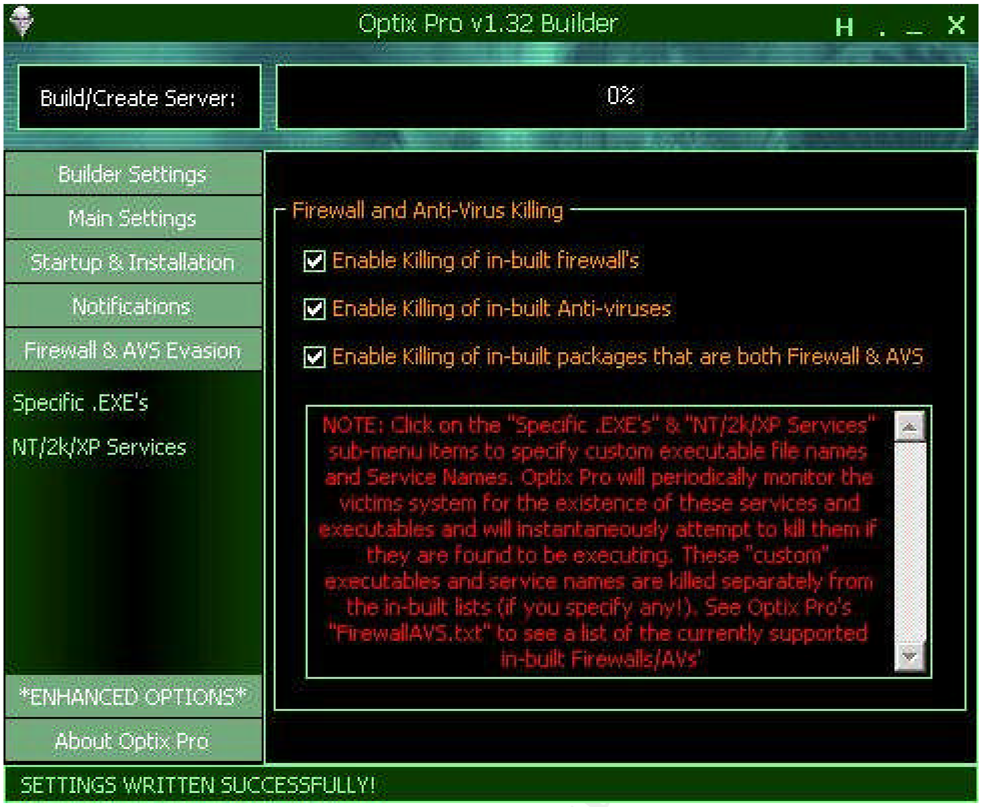
*NOTE: untick all boxes to disable startup & directory swapping!

SETTINGS WRITTEN SUCCESSFULLY!

This is a key setting as this will allow the Optix Pro server to be restarted automatically via a registry setting that it does. Even if the professors computer is rebooted or powered down the server will automatically restart due to the following registry settings being done by the above;

HKEY_LOCAL_MACHINE\SOFTWARE\Microsof\windows\CurrentVersion\run
HEKY_LOCAL_SOFTWARE\Microsoft\Windows\CurrentVersion\RunServices

John was now starting to panic a little as he had now been at this for a seemingly long period of time. He decided that he would now only activate the “Firewall & AVS Evasion” options.



Firewall and Anti-Virus Killing

☑ Enable Killing of in-built firewall's

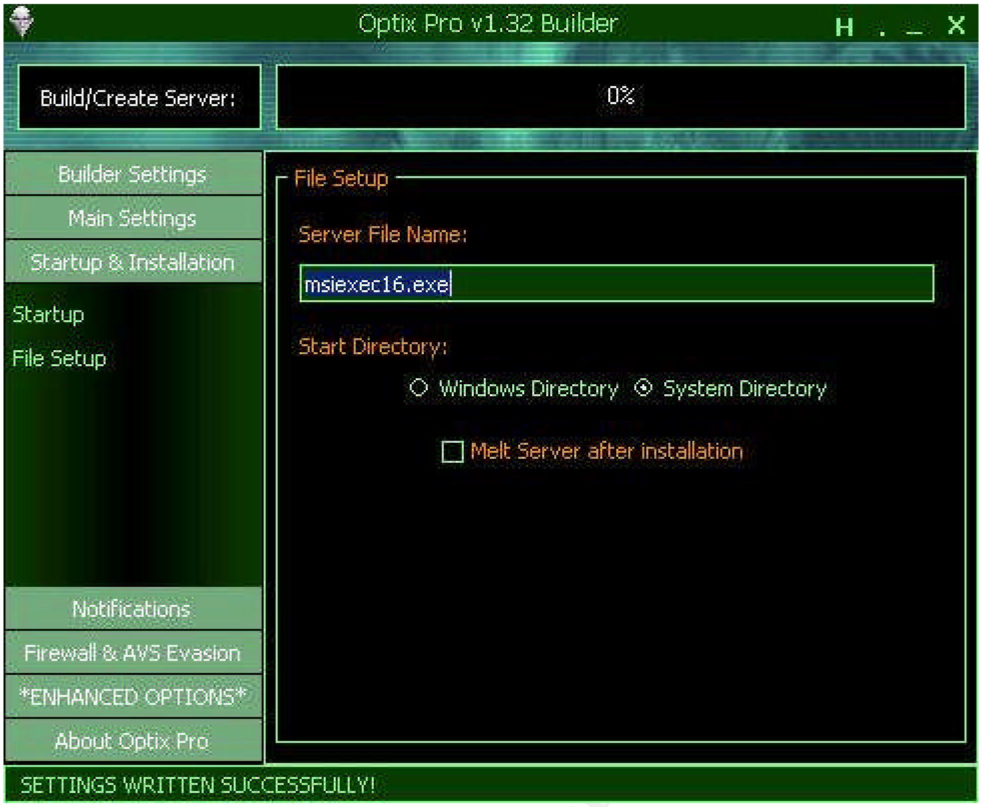
☑ Enable Killing of in-built Anti-viruses

☑ Enable Killing of in-built packages that are both Firewall & AVS

NOTE: Click on the "Specific .EXE's" & "NT/2k/XP Services" sub-menu items to specify custom executable file names and Service Names. Optix Pro will periodically monitor the victims system for the existence of these services and executables and will instantaneously attempt to kill them if they are found to be executing. These "custom" executables and service names are killed separately from the in-built lists (if you specify any!). See Optix Pro's "FirewallAVS.txt" to see a list of the currently supported in-built Firewalls/AVs'

SETTINGS WRITTEN SUCCESSFULLY!

John quickly toggled on all of the above options. Please note as well that the below noted screen shot displays the default filename that Optix Pro will show up as on the victims computer in the event of a program like Active Ports were installed. If nestat was checked all that would be seen would be the listening socket with the source and destination IP addresses, plus ports being used. This is why a program like Active Ports is so very nice.



File Setup

Server File Name: msiexec16.exe

Start Directory: ○ Windows Directory ● System Directory

□ Melt Server after installation

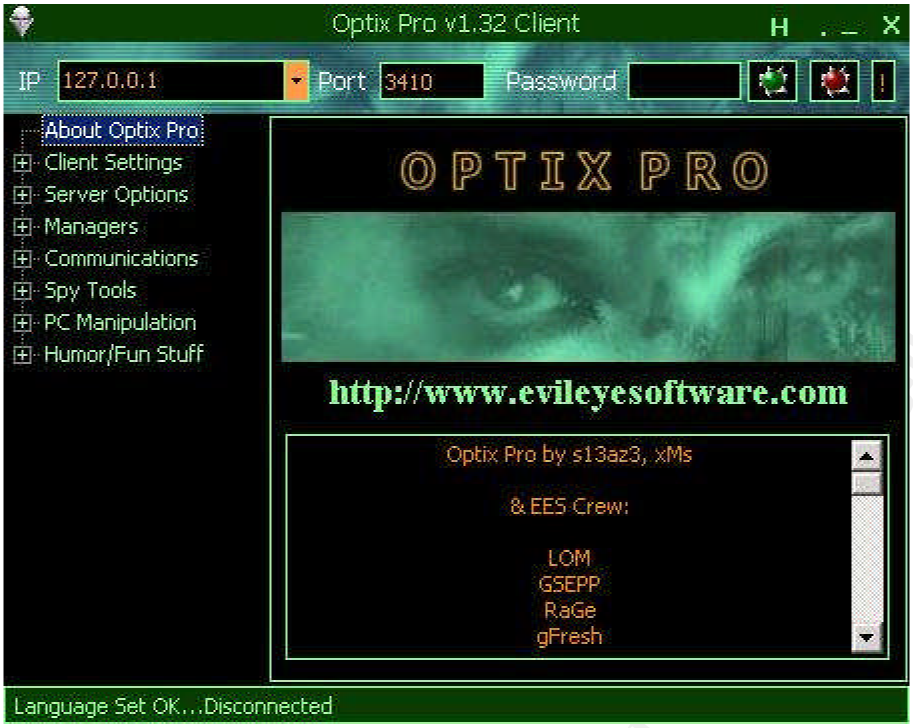
SETTINGS WRITTEN SUCCESSFULLY!

At this point John was finished, and now clicked on the "Build/Create Server" This wrote the settings to the server and also wrote it to the filename chosen and with the icon chosen earlier.

Now that the server has been built John quickly double clicks it and moves the actual server icon to the root of C drive thinking that the professor will not look there. Once the server has been activated as mentioned it does some registry entries as noted above earlier. This as previously mentioned will allow the Optix Pro server to be restarted in the event the computer is rebooted. The job now complete John very quickly leaves the professors office, and heads back to class.

John has now regained his seat with only 3 minutes having expired. Feeling quite proud he decided to quickly test that he had connectivity to the professors computer via the Optix Pro server ie: was it indeed listening on the port he had entered (TCP Port 555).

First though John had to activate his Optix Pro client by which he would control the server installed on the professors computer. John now double clicked on the client and up came the dialog box as seen below;



IP 127.0.0.1

Port 3410

Optix Pro by s13az3, xMs

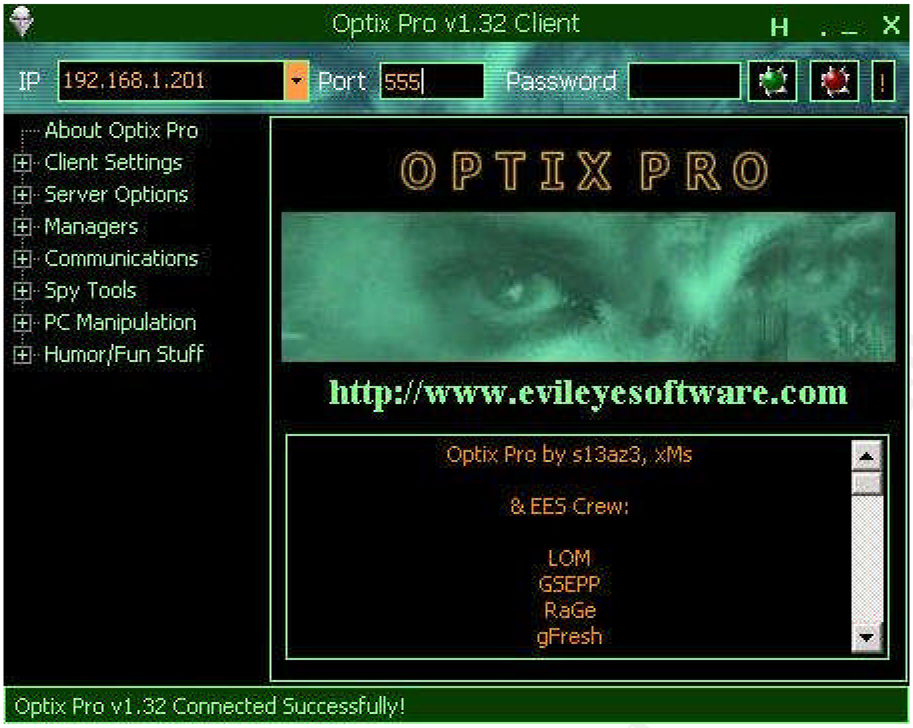
& EES Crew:

LOM
GSEPP
RaGe
gFresh

Language Set OK…Disconnected

Already knowing the professors IP address he quickly entered it and also the port of 555 vice 3410. John thought himself also quite clever by simply changing the default port for the Optix Pro server. It was also intentional on his part to pick a port which was in the "well known" range vice an ephemeral one which most trojan servers listen on.

Once finished putting the information such as the IP address and port number he then connected to the computer;



IP 192.168.1.201 Port 555 Password

Optix Pro v1.32 Connected Successfully!

Excellent! John was now connected to the professors computer while everyone around him was engrossed in the professor’s class. Please note the below noted packet trace showing the Optix client connecting to the Optix server. The trace will help clarify how the actual connection is done between the computers and also verify that the trojans transport protocol is TCP.

Packet Trace of Client Connection

```
10:37:47.817559 IP (tos 0x0, ttl 128, id 0, len 48) 192.168.1.200.1028 >
192.168.1.201.555: S [tcp sum ok]
3840986605:3840986605(0) win 65535 <mss 1460,nop,nop,sackOK> (DF)
0x0000          4500 0030 0000 4000 8006 75e6 c0a8 01c8 E..0..@...u.....
0x0010          c0a8 01c9 0404 022b e4f0 cded 0000 0000  .......+........
0x0020          7002 ffff 4530 0000 0204 05b4 0101 0402  p...E0..........
```

The above packet shows the first step of the 3 way TCP/IP handshake ie: the SYN packet which comes from John's computer and is sent to the professors comptuer.

```
10:37:47.817706 IP (tos 0x0, ttl 128, id 102, len 48) 192.168.1.201.555 >
192.168.1.200.1028: S [tcp sum ok] 1525306625:1525306625(0) ack 3840986606 win
65535 <mss 1460,nop,nop,sackOK> (DF)
0x0000          4500 0030 0066 4000 8006 7580 c0a8 01c9 E..0.f@...u.....
0x0010          c0a8 01c8 022b 0404 5aea 5501 e4f0 cdee  .....+..Z.U.....
0x0020          7012 ffff 9533 0000 0204 05b4 0101 0402  p....3..........
```

The professors computer as noted above syn/ack's back (second step in the 3 way TCP/IP handshake) because there is indeed a service listening on TCP Port 555.

```
10:37:47.817847 IP (tos 0x0, ttl 128, id 1, len 40) 192.168.1.200.1028 >
192.168.1.201.555: . [tcp sum ok] ack 1525306626 win 65535 (DF)
0x0000          4500 0028 0001 4000 8006 75ed c0a8 01c8 E..(..@...u.....
0x0010          c0a8 01c9 0404 022b e4f0 cdee 5aea 5502  .......+....Z.U.
0x0020          5010 ffff c1f7 0000 0000 0000 0000        P............
```

At this point John's computer completes the handshake with the ack packet seen above. The two computers are now ready to exchange information as the sequence numbers have now been synchonized.

```
10:37:47.819212 IP (tos 0x0, ttl 128, id 103, len 43) 192.168.1.201.555 >
192.168.1.200.1028: P [tcp sum ok]
1525306626:1525306629(3) ack 3840986606 win 65535 (DF)
0x0000          4500 002b 0067 4000 8006 7584 c0a8 01c9 E..+.g@...u.....
0x0010          c0a8 01c8 022b 0404 5aea 5502 e4f0 cdee  .....+..Z.U.....
0x0020          5018 ffff 97df 0000 200d 0a               P..........
```

The above is a psh/ack packet with no actual data in it. Though please note that in actuality there is 3 bytes of data reflected as being present in the packet headers. (the number seen after the tcp sequence nunmber of 1525306626:1525306629(3) The 3 means that there is 3 bytes of data being sent in this packet. Though as we look in the ascii content on the right we see no apparent data.

```
10:37:47.822014 IP (tos 0x0, ttl 128, id 2, len 51) 192.168.1.200.1028 >
192.168.1.201.555: P [tcp sum ok]
3840986606:3840986617(11) ack 1525306629 win 65532 (DF)
0x0000          4500 0033 0002 4000 8006 75e1 c0a8 01c8 E..3..@...u.....
0x0010          c0a8 01c9 0404 022b e4f0 cdee 5aea 5505  .......+....Z.U.
0x0020          5018 fffc 4554 0000 3032 32ac ac76 312e  P...ET..022..v1.
0x0030          320d 0a                                    2..
```

Now John's computer psh/ack's back to the professors computer an apparent 11 bytes of data. It does seem that it has perhaps sent some data as noted by the “v12” in the ascii content but it is not very conclusive. That being said according to the packet itself there are 11 bytes of data that were sent.

This is evidenced once again by the 11 in brackets following the TCP sequence number. At this time also please note the difference between the TCP sequence number on the left and the one on the right of the full colon. They do not match up! Why is this? Well if you will note there is an exact difference of 11 between both TCP sequence numbers. What this means in reality is that every byte of data being sent actually has a TCP sequence number assigned to it. This is after all TCP that is being used and this is part of the connection oriented that is TCP. In essence the “guaranteed” delivery that TCP is known for.

```
10:37:47.822736 IP (tos 0x0, ttl 128, id 104, len 85) 192.168.1.201.555 >
192.168.1.200.1028: P [tcp sum ok]
1525306629:1525306674(45) ack 3840986617 win 65524 (DF)
0x0000          4500 0055 0068 4000 8006 7559 c0a8 01c9   E..U.h@...uY....
0x0010          c0a8 01c8 022b 0404 5aea 5505 e4f0 cdf9   .....+..Z.U.....
0x0020          5018 fff4 0165 0000 3030 31ac 4f70 7469   P....e..001.Opti
0x0030          7820 5072 6f20 7631 2e33 3220 436f 6e6e   x.Pro.v1.32.Conn
0x0040          6563 7465 6420 5375 6363 6573 7366 756c   ected.Successful
0x0050          6c79 210d 0a                                ly!..
```

NOTE

It would trivial to detect this trojans usage in a college or for that matter any other environment with an IDS (intrusion detection system). A savvy security analyst would go ahead and write up some custom signatures looking for these ascii strings. Personally I would test in a lab quickly all of the well known trojans for this very reason and then write some custom signatures to detect their usage.

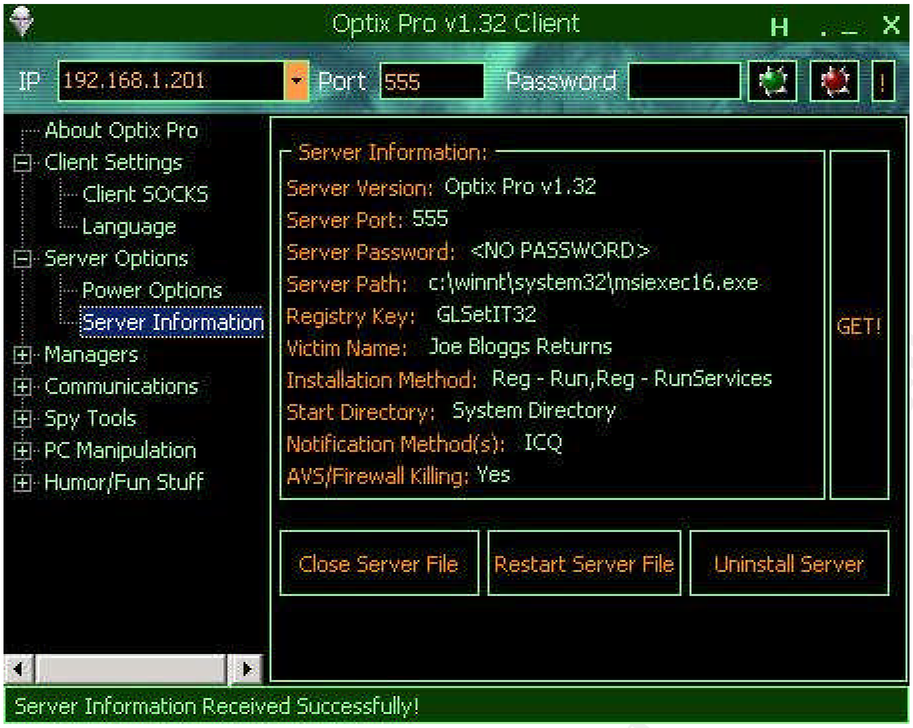
Now as seen above the professor’s computer send across 45 bytes of data to John's computer. That data as seen in the ascii content tells John that he has successfully connected to the professors computer.

```
10:37:47.945432 IP (tos 0x0, ttl 128, id 3, len 40) 192.168.1.200.1028 >
192.168.1.201.555: . [tcp sum ok] ack 1525306674 win
65487 (DF)
0x0000          4500 0028 0003 4000 8006 75eb c0a8 01c8 E..(..@...u.....
0x0010          c0a8 01c9 0404 022b e4f0 cdf9 5aea 5532   .......+....Z.U2
0x0020          5010 ffcf c1ec 0000 0000 0000 0000          P............
```

John's computer now ack's back receipt of the bytes that were just sent to him by the professors computer. This is seen in the ack number of 1525306674, which was the TCP sequence number of the previous packet noted above.

John has now successfully connected to the professor’s computer and now wants to take a peek around it to see if he can find the upcoming math exam. He knows from his previous experimenting with the Optix Pro trojan at home that the can list the victims computers drives. He now proceeds to do just that so he can try and find that math exam.

He first though clicks on server options in his Optix client to get and confirm the information he entered when he installed it on the professor’s computer. This is seen below;

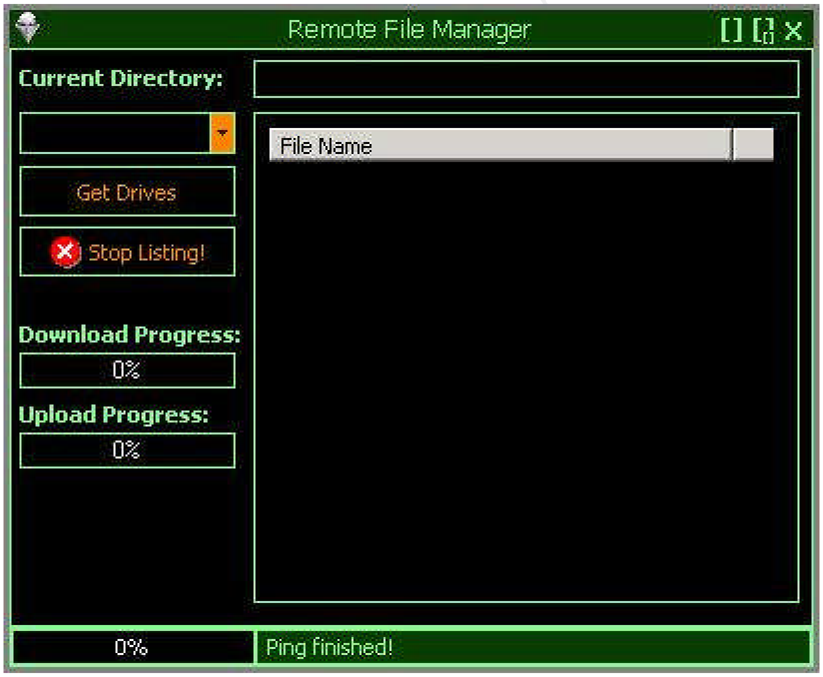


IP 192.168.1.201 Port 555 Password

Server Information:
Server Version: Optix Pro v1.32
Server Port: 555
Server Password: <NO PASSWORD>
Server Path: c:\winnt\system32\msiexec16.exe
Registry Key: GLSetIT32
Victim Name: Joe Bloggs Returns
Installation Method: Reg - Run,Reg - RunServices
Start Directory: System Directory
Notification Method(s): ICQ
AVS/Firewall Killing: Yes

Server Information Received Successfully!

Now that he has confirmed the settings he entered he clicks on the “manager” menu as seen above on the right. This will allow him to browse the professor’s computer contents.



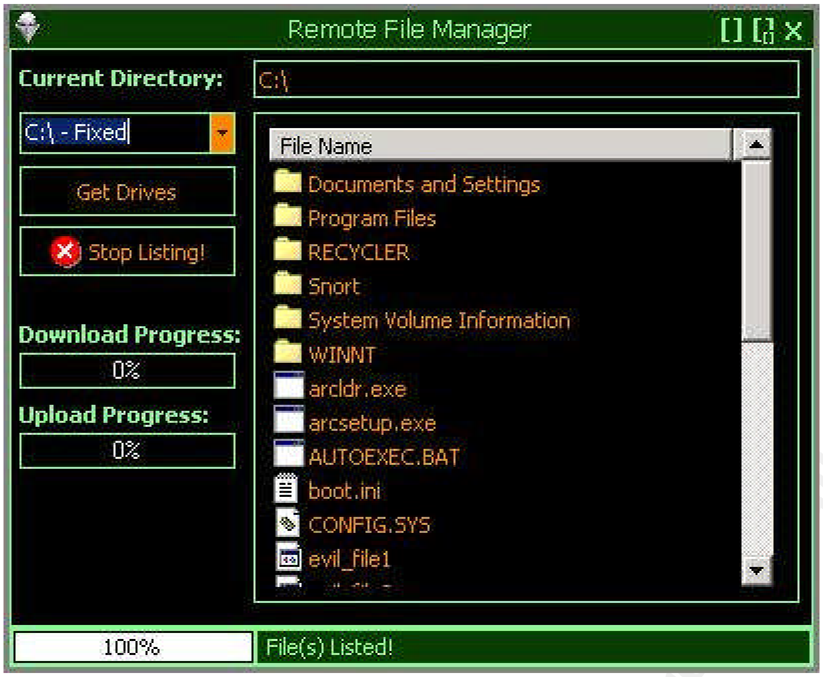
Current Directory:

Download Progress: 0%

Upload Progress: 0%

0% Ping finished!

With the above noted window he clicks on the “Get Drives” and Optix Pro on the professor’s computer dutifully sends across a list of the drives to John’s computer as seen below;



Current Directory: C:\

C:\ - Fixed

Download Progress: 0%

Upload Progress: 0%

```
Documents and Settings
Program Files
RECYCLER
Snort
System Volume Information
WINNT
arcldr.exe
arcsetup.exe
AUTOEXEC.BAT
boot.ini
CONFIG.SYS
evil_file1
```

100% File(s) Listed!

It is now simply a matter of doing some normal directory navigation for John to try and find the math exam for he has a complete listing of the professor's hard drive. In the interest of showing some of the key components of what is transpiring here I have included a packet trace showing the transfer of the drive listing and the actual transfer of the math exam itself once John has found it. This packet trace is lengthy but also helps understand what is going on in the background behind the Optix Pro gui.

Packet Trace of Directory Listing & File Transfer

Seen below are the results of John doing a listing of the drives on the professors computer. I will highlight certain area's of interest.

```
10:40:51.116317 IP (tos 0x0, ttl 128, id 4, len 46) 192.168.1.200.1028 >
192.168.1.201.555: P [tcp sum ok] 3840986617:3840986623(6) ack 1525306674 win
65487 (DF)
0x0000       4500 002e 0004 4000 8006 75e4 c0a8 01c8 E.....@...u.....
0x0010       c0a8 01c9 0404 022b e4f0 cdf9 5aea 5532  .......+....Z.U2
0x0020       5018 ffcf 51f8 0000 3030 32ac 0d0a       P...Q...002...


10:40:51.117369 IP (tos 0x0, ttl 128, id 139, len 86) 192.168.1.201.555 >
192.168.1.200.1028: P [tcp sum ok] 1525306674:1525306720(46) ack 3840986623 win
65518 (DF)
0x0000       4500 0056 008b 4000 8006 7535 c0a8 01c9 E..V..@...u5....
0x0010       c0a8 01c8 022b 0404 5aea 5532 e4f0 cdff  .....+..Z.U2....
0x0020       5018 ffee 0b8d 0000 3030 32ac 413a 5c20  P.......002.A:\.
0x0030       2d20 5265 6d6f 7661 626c 65ac 433a 5c20  -.Removable.C:\.
```

```
0x0040          2d20 4669 7865 64ac 443a 5c20 2d20 4344 -.Fixed.D:\.-.CD
0x0050          2d52 4f4d 0d0a                           -ROM..
```

Initial listing of the drives is noted above.

```
10:40:51.314571 IP (tos 0x0, ttl 128, id 5, len 40) 192.168.1.200.1028 >
192.168.1.201.555: . [tcp sum ok] ack 1525306720 win 65441 (DF)
0x0000          4500 0028 0005 4000 8006 75e9 c0a8 01c8 E..(..@...u.....
0x0010          c0a8 01c9 0404 022b e4f0 cdff 5aea 5560  .......+....Z.U`
0x0020          5010 ffa1 c1e6 0000 0000 0000 0000       P............

10:41:01.246893 IP (tos 0x0, ttl 128, id 6, len 49) 192.168.1.200.1028 >
192.168.1.201.555: P [tcp sum ok] 3840986623:3840986632(9) ack 1525306720 win
65441 (DF)
0x0000          4500 0031 0006 4000 8006 75df c0a8 01c8 E..1..@...u.....
0x0010          c0a8 01c9 0404 022b e4f0 cdff 5aea 5560  .......+....Z.U`
0x0020          5018 ffa1 b4b1 0000 3030 33ac 433a 5c0d  P.......003.C:\.
0x0030          0a                                        .
```

John’s computer is acknowledging receipt of the 86 bytes of data from the professorss computer which was the actual listing of his drive. This is represented by the above two packets.

```
10:41:01.250738 IP (tos 0x0, ttl 128, id 140, len 49) 192.168.1.201.555 >
192.168.1.200.1028: P [tcp sum ok] 1525306720:1525306729(9) ack 3840986632 win
65509 (DF)
0x0000          4500 0031 008c 4000 8006 7559 c0a8 01c9 E..1..@...uY....
0x0010          c0a8 01c8 022b 0404 5aea 5560 e4f0 ce08  .....+..Z.U`....
0x0020          5018 ffe5 e56c 0000 3030 35ac 3332 390d  P....l..005.329.
0x0030          0a                             .


10:41:01.429398 IP (tos 0x0, ttl 128, id 7, len 40) 192.168.1.200.1028 >
192.168.1.201.555: . [tcp sum ok] ack 1525306729 win 65432 (DF)
0x0000          4500 0028 0007 4000 8006 75e7 c0a8 01c8 E..(..@...u.....
0x0010          c0a8 01c9 0404 022b e4f0 ce08 5aea 5569  .......+....Z.Ui
0x0020          5010 ff98 c1dd 0000 0000 0000 0000       P............


10:41:01.429497 IP (tos 0x0, ttl 128, id 141, len 382) 192.168.1.201.555 >
192.168.1.200.1028: P [tcp sum ok] 1525306729:1525307071(342) ack 3840986632 win
65509 (DF)
0x0000          4500 017e 008d 4000 8006 740b c0a8 01c9 E..~..@...t.....
0x0010          c0a8 01c8 022b 0404 5aea 5569 e4f0 ce08  .....+..Z.Ui....
0x0020          5018 ffe5 1cd5 0000 3030 33ac 5c44 6f63  P.......003.\Doc
0x0030          756d 656e 7473 2061 6e64 2053 6574 7469 uments.and.Setti
```

```
0x0040        6e67 73ac 5c50 726f 6772 616d 2046 696c  ngs.\Program.Fil
0x0050        6573 ac5c 5245 4359 434c 4552 ac5c 536e  es.\RECYCLER.\Sn
0x0060        6f72 74ac 5c53 7973 7465 6d20 566f 6c75  ort.\System.Volu
```

(The above packet is truncated for brevities sake as it is listing the directory contents)

Now John is actively navigating through the professor’s computer as seen above in his hunt for the math exam.

```
10:41:01.629681 IP (tos 0x0, ttl 128, id 8, len 40) 192.168.1.200.1028 >
192.168.1.201.555: . [tcp sum ok] ack 1525307071 win 65090 (DF)
0x0000        4500 0028 0008 4000 8006 75e6 c0a8 01c8 E..(..@...u.....
0x0010        c0a8 01c9 0404 022b e4f0 ce08 5aea 56bf  .......+....Z.V.
0x0020        5010 fe42 c1dd 0000 0000 0000 0000       P..B..........

10:41:30.386530 IP (tos 0x0, ttl 128, id 9, len 72) 192.168.1.200.1028 >
192.168.1.201.555: P [tcp sum ok] 3840986632:3840986664(32) ack 1525307071 win
65090 (DF)
0x0000        4500 0048 0009 4000 8006 75c5 c0a8 01c8 E..H..@...u.....
0x0010        c0a8 01c9 0404 022b e4f0 ce08 5aea 56bf  .......+....Z.V.
0x0020        5018 fe42 87e3 0000 3030 33ac 433a 5c44  P..B....003.C:\D
0x0030        6f63 756d 656e 7473 2061 6e64 2053 6574 ocuments.and.Set
0x0040        7469 6e67 735c 0d0a                      tings\..


10:41:30.389513 IP (tos 0x0, ttl 128, id 142, len 48) 192.168.1.201.555 >
192.168.1.200.1028: P [tcp sum ok] 1525307071:1525307079(8) ack 3840986664 win
65477 (DF)
0x0000        4500 0030 008e 4000 8006 7558 c0a8 01c9 E..0..@...uX....
0x0010        c0a8 01c8 022b 0404 5aea 56bf e4f0 ce28  .....+..Z.V....(
0x0020        5018 ffc5 1912 0000 3030 35ac 3432 0d0a  P.......005.42..


10:41:30.572182 IP (tos 0x0, ttl 128, id 10, len 40) 192.168.1.200.1028 >
192.168.1.201.555: . [tcp sum ok] ack 1525307079 win 65082 (DF)
0x0000        4500 0028 000a 4000 8006 75e4 c0a8 01c8 E..(..@...u.....
0x0010        c0a8 01c9 0404 022b e4f0 ce28 5aea 56c7  .......+...(Z.V.
0x0020        5010 fe3a c1bd 0000 0000 0000 0000       P..:..........


10:41:30.572299 IP (tos 0x0, ttl 128, id 143, len 95) 192.168.1.201.555 >
192.168.1.200.1028: P [tcp sum ok] 1525307079:1525307134(55) ack 3840986664 win
65477 (DF)
0x0000        4500 005f 008f 4000 8006 7528 c0a8 01c9  E.._..@...u(....
0x0010        c0a8 01c8 022b 0404 5aea 56c7 e4f0 ce28  .....+..Z.V....(
0x0020        5018 ffc5 53ee 0000 3030 33ac 2e2e ac5c   P...S...003....\
0x0030           4164 6d69 6e69 7374 7261 746f 72ac 5c41  Administrator.\A
0x0040           6c6c 2055 7365 7273 ac5c 4465 6661 756c  ll.Users.\Defaul
0x0050           7420 5573 6572 7465 726d 2323 650d 0a     t.Userterm##e..


10:41:30.772451 IP (tos 0x0, ttl 128, id 11, len 40) 192.168.1.200.1028 >
192.168.1.201.555: . [tcp sum ok] ack 1525307134 win 65027 (DF)
0x0000           4500 0028 000b 4000 8006 75e3 c0a8 01c8  E..(..@...u.....
0x0010           c0a8 01c9 0404 022b e4f0 ce28 5aea 56fe  .......+...(Z.V.
0x0020           5010 fe03 c1bd 0000 0000 0000 0000       P.............


10:41:33.566733 IP (tos 0x0, ttl 128, id 12, len 86) 192.168.1.200.1028 >
192.168.1.201.555: P [tcp sum ok] 3840986664:3840986710(46) ack 1525307134 win
65027 (DF)
0x0000           4500 0056 000c 4000 8006 75b4 c0a8 01c8  E..V..@...u.....
0x0010           c0a8 01c9 0404 022b e4f0 ce28 5aea 56fe  .......+...(Z.V.
0x0020           5018 fe03 9ddc 0000 3030 33ac 433a 5c44  P.......003.C:\D
0x0030           6f63 756d 656e 7473 2061 6e64 2053 6574  ocuments.and.Set
0x0040           7469 6e67 735c 4164 6d69 6e69 7374 7261  tings\Administra
0x0050           746f 725c 0d0a                           tor\..


10:41:33.570460 IP (tos 0x0, ttl 128, id 144, len 49) 192.168.1.201.555 >
192.168.1.200.1028: P [tcp sum ok] 1525307134:1525307143(9) ack 3840986710 win
65431 (DF)
0x0000           4500 0031 0090 4000 8006 7555 c0a8 01c9  E..1..@...uU....
0x0010           c0a8 01c8 022b 0404 5aea 56fe e4f0 ce56  .....+..Z.V....V
0x0020           5018 ff97 e9c9 0000 3030 35ac 3137 350d  P.......005.175.
0x0030           0a                                .


10:41:33.676716 IP (tos 0x0, ttl 128, id 13, len 40) 192.168.1.200.1028 >
192.168.1.201.555: . [tcp sum ok] ack 1525307143 win 65018 (DF)
0x0000           4500 0028 000d 4000 8006 75e1 c0a8 01c8  E..(..@...u.....
0x0010           c0a8 01c9 0404 022b e4f0 ce56 5aea 5707  .......+...VZ.W.
0x0020           5010 fdfa c18f 0000 0000 0000 0000       P.............


10:41:33.676800 IP (tos 0x0, ttl 128, id 145, len 228) 192.168.1.201.555 >
192.168.1.200.1028: P [tcp sum ok] 1525307143:1525307331(188) ack 3840986710 win
65431 (DF)
0x0000           4500 00e4 0091 4000 8006 74a1 c0a8 01c9  E.....@...t.....
0x0010           c0a8 01c8 022b 0404 5aea 5707 e4f0 ce56  .....+..Z.W....V
0x0020           5018 ff97 fd59 0000 3030 33ac 2e2e ac5c  P....Y..003....\
0x0030           4170 706c 6963 6174 696f 6e20 4461 7461  Application.Data
```

```
0x0040          ac5c 436f 6f6b 6965 73ac 5c44 6573 6b74  .\Cookies.\Deskt
0x0050          6f70 ac5c 4661 766f 7269 7465 73ac 5c4c  op.\Favorites.\L
0x0060          6f63 616c 2053 6574 7469 6e67 73ac 5c4d  ocal.Settings.\M
0x0070          7920 446f 6375 6d65 6e74 73ac 5c4e 6574  y.Documents.\Net
0x0080          486f 6f64 ac5c 5072 696e 7448 6f6f 64ac  Hood.\PrintHood.
```

(This packet is truncated for brevities sake)

```
10:41:33.877007 IP (tos 0x0, ttl 128, id 14, len 40) 192.168.1.200.1028 >
192.168.1.201.555: . [tcp sum ok] ack 1525307331 win 64830 (DF)
0x0000          4500 0028 000e 4000 8006 75e0 c0a8 01c8  E..(..@...u.....
0x0010          c0a8 01c9 0404 022b e4f0 ce56 5aea 57c3  .......+...VZ.W.
0x0020          5010 fd3e c18f 0000 0000 0000 0000       P..>..........


10:41:36.219504 IP (tos 0x0, ttl 128, id 15, len 94) 192.168.1.200.1028 >
192.168.1.201.555: P [tcp sum ok] 3840986710:3840986764(54) ack 1525307331 win
64830 (DF)
0x0000          4500 005e 000f 4000 8006 75a9 c0a8 01c8  E..^..@...u.....
0x0010          c0a8 01c9 0404 022b e4f0 ce56 5aea 57c3  .......+...VZ.W.
0x0020          5018 fd3e 010a 0000 3030 33ac 433a 5c44  P..>....003.C:\D
0x0030          6f63 756d 656e 7473 2061 6e64 2053 6574  ocuments.and.Set
0x0040          7469 6e67 735c 4164 6d69 6e69 7374 7261  tings\Administra
0x0050          746f 725c 4465 736b 746f 705c 0d0a       tor\Desktop\..


10:41:36.223139 IP (tos 0x0, ttl 128, id 146, len 49) 192.168.1.201.555 >
192.168.1.200.1028: P [tcp sum ok] 1525307331:1525307340(9) ack 3840986764 win
65377 (DF)
0x0000          4500 0031 0092 4000 8006 7553 c0a8 01c9  E..1..@...uS....
0x0010          c0a8 01c8 022b 0404 5aea 57c3 e4f0 ce8c  .....+..Z.W.....
0x0020          5018 ff61 ea0b 0000 3030 35ac 3230 330d  P..a....005.203.
0x0030          0a                                       .


10:41:36.380682 IP (tos 0x0, ttl 128, id 16, len 40) 192.168.1.200.1028 >
192.168.1.201.555: . [tcp sum ok] ack 1525307340 win 64821 (DF)
0x0000          4500 0028 0010 4000 8006 75de c0a8 01c8  E..(..@...u.....
0x0010          c0a8 01c9 0404 022b e4f0 ce8c 5aea 57cc  .......+....Z.W.
0x0020          5010 fd35 c159 0000 0000 0000 0000       P..5.Y........


10:41:36.380759 IP (tos 0x0, ttl 128, id 147, len 256) 192.168.1.201.555 >
192.168.1.200.1028: P [tcp sum ok] 1525307340:1525307556(216) ack 3840986764 win
65377 (DF)
0x0000          4500 0100 0093 4000 8006 7483 c0a8 01c9  E.....@...t.....
```

```
0x0010          c0a8 01c8 022b 0404 5aea 57cc e4f0 ce8c    .....+..Z.W.....
0x0020          5018 ff61 69b9 0000 3030 33ac 2e2e ac5c    P..ai...003....\
0x0030          436c 6965 6e74 536f 6674 7761 7265 ac5c    ClientSoftware.\
0x0040          446f 6375 6d65 6e74 6174 696f 6eac 5c45    Documentation.\E
```

(This packet is truncated for brevities sake as it lists the contents of the drive)

```
10:41:36.580978 IP (tos 0x0, ttl 128, id 17, len 40) 192.168.1.200.1028 >
192.168.1.201.555: . [tcp sum ok] ack 1525307556 win 64605 (DF)
0x0000          4500 0028 0011 4000 8006 75dd c0a8 01c8 E..(..@...u.....
0x0010          c0a8 01c9 0404 022b e4f0 ce8c 5aea 58a4    .......+....Z.X.
0x0020          5010 fc5d c159 0000 0000 0000 0000         P..].Y........


10:41:44.946462 IP (tos 0x0, ttl 128, id 18, len 104) 192.168.1.200.1028 >
192.168.1.201.555: P [tcp sum ok] 3840986764:3840986828(64) ack 1525307556 win
64605 (DF)
0x0000          4500 0068 0012 4000 8006 759c c0a8 01c8 E..h..@...u.....
0x0010          c0a8 01c9 0404 022b e4f0 ce8c 5aea 58a4    .......+....Z.X.
0x0020          5018 fc5d d9dc 0000 3030 33ac 433a 5c44  P..]....003.C:\D
0x0030          6f63 756d 656e 7473 2061 6e64 2053 6574 ocuments.and.Set
0x0040          7469 6e67 735c 4164 6d69 6e69 7374 7261 tings\Administra
0x0050          746f 725c 4465 736b 746f 705c 6d61 7468  tor\Desktop\math
0x0060          5f65 7861 6d5c 0d0a                          _exam\..
```

John now gets his first indication that the math exam is indeed on the professors computer as seen in the ascii content above.

```
10:41:44.949983 IP (tos 0x0, ttl 128, id 148, len 48) 192.168.1.201.555 >
192.168.1.200.1028: P [tcp sum ok] 1525307556:1525307564(8) ack 3840986828 win
65313 (DF)
0x0000          4500 0030 0094 4000 8006 7552 c0a8 01c9 E..0..@...uR....
0x0010          c0a8 01c8 022b 0404 5aea 58a4 e4f0 cecc    .....+..Z.X.....
0x0020          5018 ff21 1a29 0000 3030 35ac 3136 0d0a  P..!.)..005.16..


10:41:45.093504 IP (tos 0x0, ttl 128, id 19, len 40) 192.168.1.200.1028 >
192.168.1.201.555: . [tcp sum ok] ack 1525307564 win 64597 (DF)
0x0000          4500 0028 0013 4000 8006 75db c0a8 01c8 E..(..@...u.....
0x0010          c0a8 01c9 0404 022b e4f0 cecc 5aea 58ac    .......+....Z.X.
0x0020          5010 fc55 c119 0000 0000 0000 0000         P..U..........


10:41:45.093589 IP (tos 0x0, ttl 128, id 149, len 69) 192.168.1.201.555 >
192.168.1.200.1028: P [tcp sum ok] 1525307564:1525307593(29) ack 3840986828 win
65313 (DF)
```

```
0x0000           4500 0045 0095 4000 8006 753c c0a8 01c9 E..E..@...u<....
0x0010           c0a8 01c8 022b 0404 5aea 58ac e4f0 cecc  .....+..Z.X.....
0x0020           5018 ff21 cf0a 0000 3030 33ac 2e2e ac6d  P..!....003....m
0x0030           6174 685f 6578 616d 2e74 7874 7465 726d ath_exam.txtterm
0x0040           2323 650d 0a                              ##e..

10:41:45.293763 IP (tos 0x0, ttl 128, id 20, len 40) 192.168.1.200.1028 >
192.168.1.201.555: . [tcp sum ok] ack 1525307593 win 64568 (DF)
0x0000           4500 0028 0014 4000 8006 75da c0a8 01c8 E..(..@...u.....
0x0010           c0a8 01c9 0404 022b e4f0 cecc 5aea 58c9  .......+....Z.X.
0x0020           5010 fc38 c119 0000 0000 0000 0000       P..8..........


10:42:05.358300 IP (tos 0x0, ttl 128, id 21, len 46) 192.168.1.200.1028 >
192.168.1.201.555: P [tcp sum ok] 3840986828:3840986834(6) ack 1525307593 win
64568 (DF)
0x0000           4500 002e 0015 4000 8006 75d3 c0a8 01c8 E.....@...u.....
0x0010           c0a8 01c9 0404 022b e4f0 cecc 5aea 58c9  .......+....Z.X.
0x0020           5018 fc38 4e22 0000 3033 35ac 0d0a       P..8N"..035...


10:42:05.361405 IP (tos 0x0, ttl 128, id 151, len 46) 192.168.1.201.555 >
192.168.1.200.1028: P [tcp sum ok] 1525307593:1525307599(6) ack 3840986834 win
65307 (DF)
0x0000           4500 002e 0097 4000 8006 7551 c0a8 01c9 E.....@...uQ....
0x0010           c0a8 01c8 022b 0404 5aea 58c9 e4f0 ced2  .....+..Z.X.....
0x0020           5018 ff1b 4a39 0000 3033 36ac 0d0a       P...J9..036...


10:42:05.365387 IP (tos 0x0, ttl 128, id 22, len 48) 192.168.1.200.1029 >
192.168.1.201.501: S [tcp sum ok] 3905336144:3905336144(0) win 65535 <mss
1460,nop,nop,sackOK> (DF)
0x0000           4500 0030 0016 4000 8006 75d0 c0a8 01c8 E..0..@...u.....
0x0010           c0a8 01c9 0405 01f5 e8c6 b350 0000 0000  ...........P....
0x0020           7002 ffff 5c2c 0000 0204 05b4 0101 0402  p...\,..........


10:42:05.365495 IP (tos 0x0, ttl 128, id 152, len 48) 192.168.1.201.501 >
192.168.1.200.1029: S [tcp sum ok] 1589619150:1589619150(0) ack 3905336145 win
65535 <mss 1460,nop,nop,sackOK> (DF)
0x0000           4500 0030 0098 4000 8006 754e c0a8 01c9 E..0..@...uN....
0x0010           c0a8 01c8 01f5 0405 5ebf a9ce e8c6 b351  ........^......Q
0x0020           7012 ffff 538d 0000 0204 05b4 0101 0402  p...S...........
```

```
10:42:05.365636 IP (tos 0x0, ttl 128, id 23, len 40) 192.168.1.200.1029 >
192.168.1.201.501: . [tcp sum ok] ack 1589619151 win 65535 (DF)
0x0000          4500 0028 0017 4000 8006 75d7 c0a8 01c8 E..(..@...u.....
0x0010          c0a8 01c9 0405 01f5 e8c6 b351 5ebf a9cf ...........Q^...
0x0020          5010 ffff 8051 0000 0000 0000 0000      P....Q........
```

```
10:42:05.366378 IP (tos 0x0, ttl 128, id 24, len 121) 192.168.1.200.1029 >
192.168.1.201.501: P [tcp sum ok] 3905336145:3905336226(81) ack 1589619151 win
65535 (DF)
0x0000          4500 0079 0018 4000 8006 7585 c0a8 01c8 E..y..@...u.....
0x0010          c0a8 01c9 0405 01f5 e8c6 b351 5ebf a9cf ...........Q^...
0x0020          5018 ffff 9818 0000 496e 666f 4f6e ac43 P.......InfoOn.C
0x0030          3a5c 446f 6375 6d65 6e74 7320 616e 6420 :\Documents.and.
0x0040          5365 7474 696e 6773 5c41 646d 696e 6973 Settings\Adminis
0x0050          7472 6174 6f72 5c44 6573 6b74 6f70 5c6d trator\Desktop\m
0x0060          6174 685f 6578 616d 5c6d 6174 685f 6578 ath_exam\math_ex
0x0070          616d 2e74 7874 ac0d 0a                    am.txt...
```

John has now gone to the actual desktop of the professor's computer and notes the math exam is indeed there as seen in the above packet.

```
10:42:05.366805 IP (tos 0x0, ttl 128, id 153, len 43) 192.168.1.201.501 >
192.168.1.200.1029: P [tcp sum ok] 1589619151:1589619154(3) ack 3905336226 win
65454 (DF)
0x0000          4500 002b 0099 4000 8006 7552 c0a8 01c9 E..+..@...uR....
0x0010          c0a8 01c8 01f5 0405 5ebf a9cf e8c6 b3a2 ........^.......
0x0020          5018 ffae 5639 0000 200d 0a          P...V9.....
```

```
10:42:05.523463 IP (tos 0x0, ttl 128, id 25, len 40) 192.168.1.200.1028 >
192.168.1.201.555: . [tcp sum ok] ack 1525307599 win 64562 (DF)
0x0000          4500 0028 0019 4000 8006 75d5 c0a8 01c8 E..(..@...u.....
0x0010          c0a8 01c9 0404 022b e4f0 ced2 5aea 58cf .......+....Z.X.
0x0020          5010 fc32 c113 0000 0000 0000 0000      P..2..........
```

```
10:42:05.523490 IP (tos 0x0, ttl 128, id 26, len 40) 192.168.1.200.1029 >
192.168.1.201.501: . [tcp sum ok] ack 1589619154 win 65532 (DF)
0x0000          4500 0028 001a 4000 8006 75d4 c0a8 01c8 E..(..@...u.....
0x0010          c0a8 01c9 0405 01f5 e8c6 b3a2 5ebf a9d2 ............^...
0x0020          5010 fffc 8000 0000 0000 0000 0000      P.............
```

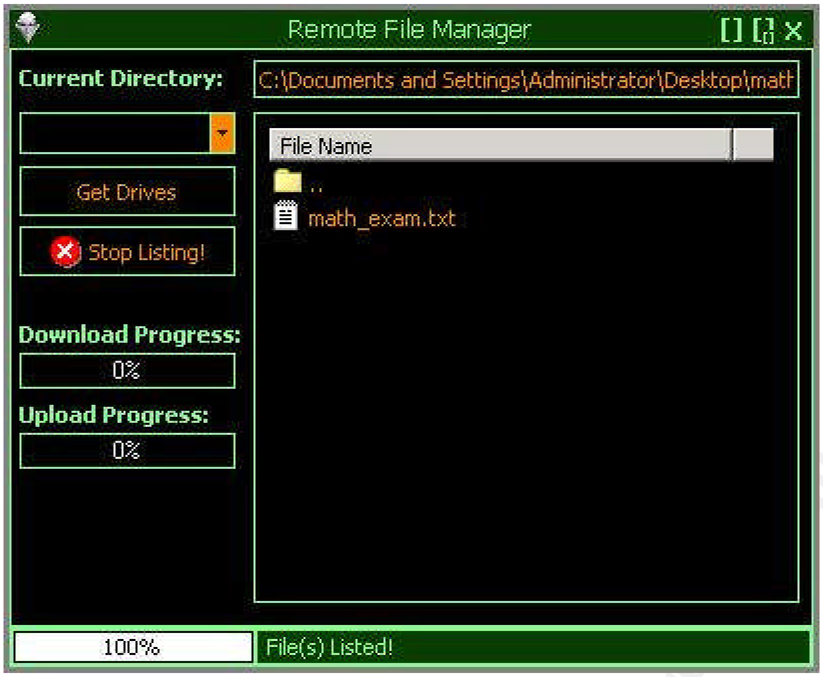
```
10:42:05.523567 IP (tos 0x0, ttl 128, id 154, len 56) 192.168.1.201.501 >
192.168.1.200.1029: P [tcp sum ok] 1589619154:1589619170(16) ack 3905336226 win
65454 (DF)
0x0000          4500 0038 009a 4000 8006 7544 c0a8 01c9 E..8..@...uD....
0x0010          c0a8 01c8 01f5 0405 5ebf a9d2 e8c6 b3a2  ........^.......
0x0020          5018 ffae cb3b 0000 4669 6c65 5369 7a65  P....;..FileSize
0x0030          4973 ac33 31ac 0d0a                          Is.31...
```

```
10:42:05.525180 IP (tos 0x0, ttl 128, id 27, len 55) 192.168.1.200.1029 >
192.168.1.201.501: P [tcp sum ok] 3905336226:3905336241(15) ack 1589619170 win
65516 (DF)
0x0000          4500 0037 001b 4000 8006 75c4 c0a8 01c8 E..7..@...u.....
0x0010          c0a8 01c9 0405 01f5 e8c6 b3a2 5ebf a9e2  ...........^...
0x0020          5018 ffec 5d88 0000 5365 6e64 6d65 6461  P...]...Sendmeda
0x0030          6461 7461 ac0d 0a                           data...
```

John now decides to download the math exam to his computer as seen by the command reflected above in the ascii content.

```
10:42:05.526606 IP (tos 0x0, ttl 128, id 155, len 71) 192.168.1.201.501 >
192.168.1.200.1029: P [tcp sum ok] 1589619170:1589619201(31) ack 3905336241 win
65439 (DF)
0x0000          4500 0047 009b 4000 8006 7534 c0a8 01c9 E..G..@...u4....
0x0010          c0a8 01c8 01f5 0405 5ebf a9e2 e8c6 b3b1  ........^.......
0x0020          5018 ff9f 5fc9 0000 5468 6973 2069 7320  P..._...This.is.
0x0030          7468 6520 6669 6374 696f 6e61 6c20 6d61 the.fictional.ma
0x0040          7468 2065 7861 6d                       th.exam
```

Noted above is the “actual” math exam itself. This was simply a file I created in notepad and entered the above noted text in it. Please see the two below noted screen shots that show the Optix Pro client both finding the file and then downloading it.



Current Directory: C:\Documents and Settings\Administrator\Desktop\math

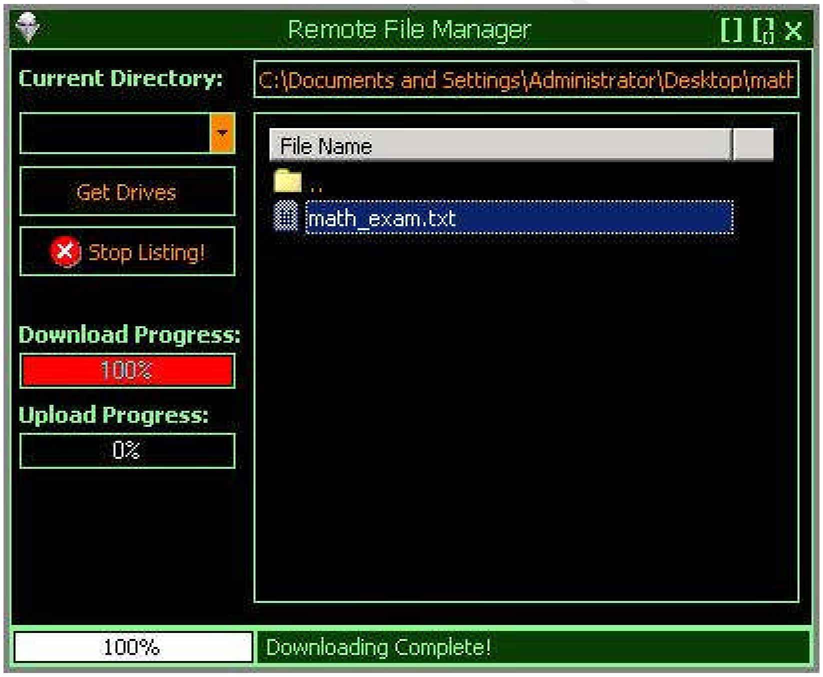
..
math_exam.txt

Download Progress: 0%

Upload Progress: 0%

100% File(s) Listed!

The above shows that Optix Pro client (John's computer) has successfully navigated the professor's computer and found the file.



Current Directory: C:\Documents and Settings\Administrator\Desktop\math

..
math_exam.txt

Download Progress: 100%

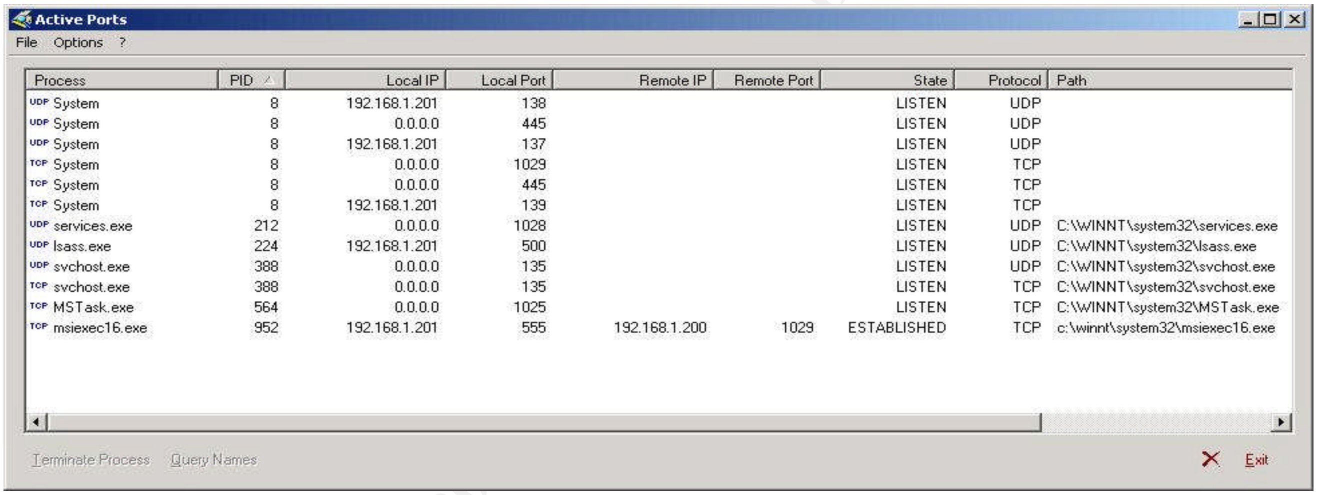
Upload Progress: 0%

100% Downloading Complete!

The above screen shot of the Optix Pro client on John's computer now shows that he successfully downloaded the math exam. This is also reflected in the above noted packet.

At this point in time John was quite happy that his scheme had paid off. He had not given any thought though to perhaps going back and deleting the Optix Pro server on the professor's computer. This was a compromised computer he wanted continued access too. Not being an overly savvy computer user security-wise it did not occur to him that he may get caught. So John in reality thought he would have continued access to the computer with little worry of being caught, or needing to do anything to hide this trail. John at the beginning when installing the Optix Pro server simply hid the icon which was the now built server on the root of C drive and thought that enough to hide his tracks.

He probably would not have been caught either had it not been for the professor being a little security conscious. We will see later how the professor unwittingly foiled John's efforts. Please see a couple of other screen shots below, which show the trojan itself. This is how it might have been viewed had the professor had something with, which to monitor his active sockets. A program such, as ActivePorts in this case.



Process	PID	Local IP	Local Port	Remote IP	Remote Port	State	Protocol	Path
UDP System	8	192.168.1.201	138			LISTEN	UDP	
UDP System	8	0.0.0.0	445			LISTEN	UDP	
UDP System	8	192.168.1.201	137			LISTEN	UDP	
TCP System	8	0.0.0.0	1029			LISTEN	TCP	
TCP System	8	0.0.0.0	445			LISTEN	TCP	
TCP System	8	192.168.1.201	139			LISTEN	TCP	
UDP services.exe	212	0.0.0.0	1028			LISTEN	UDP	C:\WINNT\system32\services.exe
UDP lsass.exe	224	192.168.1.201	500			LISTEN	UDP	C:\WINNT\system32\lsass.exe
UDP svchost.exe	388	0.0.0.0	135			LISTEN	UDP	C:\WINNT\system32\svchost.exe
TCP svchost.exe	388	0.0.0.0	135			LISTEN	TCP	C:\WINNT\system32\svchost.exe
TCP MSTask.exe	564	0.0.0.0	1025			LISTEN	TCP	C:\WINNT\system32\MSTask.exe
TCP msiexec16.exe	952	192.168.1.201	555	192.168.1.200	1029	ESTABLISHED	TCP	c:\winnt\system32\msiexec16.exe

We can see in the above screen shot the professor's computers IP address of 192.168.1.201 listening on port 555 and has an established connection with 192.168.1.200 on its port 1029. Both ports are TCP as well. What is also very nice about this program is that it will map the listening process path for you. In this case it tells us that msiexec16.exe is in the system32 folder. A very nice feature indeed! Activeports will also allow you to kill any active socket as seen on the bottom left half of the screen shot.

It should be noted here as well that the normal perimeter hardening of the college was completely circumvented by the fact that John had physical access to the computer. With this physical access in mind the router's ACL's, and application layer firewall intelligence could do nothing to prevent John's attack. Too often the avenue of physical access to a computer is overlooked. You can have all of the sophisticated protection you want. If the door is unlocked and the computer is there unguarded by even a simple door lock then all of that very expensive hardware and software is now useless. The lesson to be learned from this scenario is that you must also physically safeguard access to your computers.

Had this college had some intrusion detection systems strategically placed then the trojan infection would have been caught quite quickly. Some intrusion detection systems have signatures which will fire on a connection which originates and ends with an ephemeral port ie: P2P software does this as do trojans. Remember the default port setting of 3410 for the trojan server? That is an ephemeral port, and the Optix Pro client would of also initiated communications on an ephemeral port as well. Not to mention that in reality the topology used in this scenario is not a very secure one. Though please note that there really are colleges and universities out there who have a similar topology.

NOTE

The trojan server has a great deal of functionality built into as was displayed a little bit by the above noted scenario. This trojan though has a great deal more functionality built into it such as the ability to kill a great deal of known firewall, and anti-virus software programs. Also it will kill some well known anti-trojan utilities. Interestingly as well it allows one the ability to simply enter the name of any other executable as well. To test out this feature I went ahead and put in via the trojan server "activeports.exe" which will show up in the process list of the computer. Optix Pro killed the process once I had entered it into the server and Optix Pro then found this process listed in the computers process list.

This is a very powerful feature and one which makes this trojan so dangerous. Also note that this trojan will check the process list every 30 seconds and once again kill any anti-virus, firewall, or anti-trojan it finds in it's kill list. This is why even upon reboot and the trojan restarts itself due to its registry entry it will also kill the newly activated computer defense mechanisms. Rather impressive I think. Also please note that this trojan was coded in Delphi as is most every other trojan. This is largely due to the fact that programming in Delphi is relatively simple unlike say coding in C or ASM.

Due to page constraints all of the Optix Pro trojans capabilities were not used. There are a great many others as alluded to above. A key logger, ftp server, among other features is built into this program. All of them are functional and were tested by myself outside of the scope of this paper. I tested the parts of this trojan out that meshed well with the scenario that I was using thereby giving a realistic usage example of the trojan in action.

The Incident Handling Process

Preparation

The college that this trojan infection took place in is a small town college. This college like many out there does not have a dedicated security staff, but rather a very busy system administration team. All of the system administrators also have the dual task of doing security as well. What is meant by that is they also do up all ACL's on the router and are in charge of the firewall itself as well. No IDS is in place at this college due both to lack of funds and trained staff to maintain it. The reason I know so much about this fictional college is that a sys admin there is a friend of mine, and that is how I end up getting involved in the incident handling process.

Due to this lack of dedicated security staff the sys admin staff are very much over worked and not as up to speed on network security as they should be. This is through no fault of their own of course as their main duties are system administration, and a lack of funds on the college's part to hire full time computer security staff.

With this staff scenario in mind the normal network counter measures are in place. By that I mean a border router with ACL's in place to deny all traffic inbound with the exception of the allowed services being offered by the college. Those services being a web server, secure shell acess, and normal mail services as well as webmail. All other ports are denied inbound.

Behind the border router is an application layer firewall to help prevent and catch buffer overflows and format string attacks amongst others. This is there to help protect the services that are being offered by the college and of course must remain accessible from the exterior ie: the internet.

The mail servers themselves which did not appear in my network datagram for they are not really germane to this exploit also have an email stripping attachment solution to strip off known trouble attachments. One's such as .bat .pif and .scr among others. Please note as well that these servers reside in a DMZ. Residing behind the router as seen in the network diagram is the main switch which in turns feeds the other college department switches. This is not an overly secure network in design but does offer some protection to the most common threats.

With all of the IT staff being system administrators by trade there is no formal training that they have taken in security, nor incident handling. The odd time that a faculty person had a virus or other issue which was not handled by on board software an admin was dispatched to help. Due to this lack of organization and formal policy in place there was a rather chaotic and unorganized approach to any incidents past or present.

Identification

John started his whole adventure on Monday morning by going, and confirming that the professor still kept his office open during classes. Having that key piece of information in hand John was ready to actually go, and infect the professor's computer the next day. Prior to John physically doing active reconnaissance of the professors office he has spend several weeks playing around with Optix Pro. It was during his experimentation with Optix Pro that he dreamt up his scheme of infecting his professor's computer. This is why John spent so much time practicing, and playing with Optix Pro. He wanted to get very familiar with it, and by extension be able to quickly install it on someone's computer.

After John had confirmed the empty and open office status of his math professor he was now ready to go ahead, and infect his computer the next day. On Tuesday morning he attended the professor's class, and slipped out after about ten minutes had elapsed. Enough time he thought for the professor and the students to be engrossed in that days lecture.

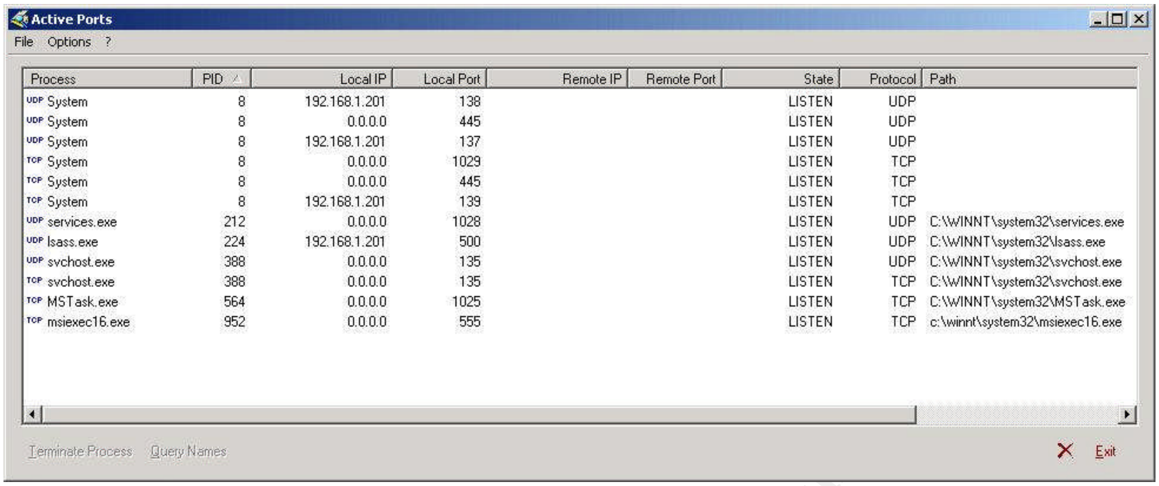
After a quick walk past the empty office and seeing no one about in the hallway he quickly entered, and sat down. It only took John a couple of minutes to install, and configure the trojan server. John simply plugged in his USB stick into the computer and copied over the trojan server onto the professor's computer. After he copied it over he simply as mentioned went through several key menu's for configuration.
(This actual exploiting of the system was covered in detail earlier in the section titled "exploiting the system")Once done this he went right back to class and sat down. All told John was away from class no more then five minutes.

This is where a key area of protection failed for the college. The anti-virus solution on the computer was able to be disabled by John. This should not be the case for this very same reason. It is also however how the professor himself was able to get the ball rolling on the upcoming investigation.

After class that morning the professor returned to this office to attend to his normal duties. Issue's such, as grading student assignments plus other daily administration. He checked his email and downloaded that mornings email to his computer. One of the emails was from a colleague who had sent him an attachment. The professor remembered the system administrators saying to always scan an attachment prior to opening it. With this in mind the professor saved the document to his desktop and right clicked on it to use the anti-virus program to scan it. Problem was though he noted the anti-virus program would not start up! He was not quite sure what to do, but remembered he should not open up the attachment as he could not scan it for viruses. The professor actually had paid attention when the system administrators gave the college staff a quick briefing on computer security.

With him being unable to scan the file the professor placed a call to the system administration office for some help. Within a couple of minutes one of the admins was in the professor's office. He also tried what the professor had and with similar results. Next the admin checked the task bar and noticed that the icon for the anti-virus solution was not there. Odd he thought, but it must of crashed was his reasoning. He then rebooted the computer, and saw that the anti-virus icon was back in the taskbar. Once the computer had finished rebooting he then went to scan the document. This had all taken about a minute or so of time.

The same problem though occurred. He was not able to get the program to come up. Now the admin went to do it via the program menu, but also noticed the icon was once again gone in the taskbar. This was not making any sense to the admin now. No way should the anti-virus program crash again so quickly if at all. Now curious the admin installed on the professor's computer a program called Activeports. This would help the admin see what was listening on the computer. In essence what was working program wise for the anti-virus program should be listed there. Listed below in the screen shot is what the admin saw.



Process	PID	Local IP	Local Port	Remote IP	Remote Port	State	Protocol	Path
UDP System	8	192.168.1.201	138			LISTEN	UDP	
UDP System	8	0.0.0.0	445			LISTEN	UDP	
UDP System	8	192.168.1.201	137			LISTEN	UDP	
TCP System	8	0.0.0.0	1029			LISTEN	TCP	
TCP System	8	0.0.0.0	445			LISTEN	TCP	
TCP System	8	192.168.1.201	139			LISTEN	TCP	
UDP services.exe	212	0.0.0.0	1028			LISTEN	UDP	C:\WINNT\system32\services.exe
UDP lsass.exe	224	192.168.1.201	500			LISTEN	UDP	C:\WINNT\system32\lsass.exe
UDP svchost.exe	388	0.0.0.0	135			LISTEN	UDP	C:\WINNT\system32\svchost.exe
TCP svchost.exe	388	0.0.0.0	135			LISTEN	TCP	C:\WINNT\system32\svchost.exe
TCP MSTask.exe	564	0.0.0.0	1025			LISTEN	TCP	C:\WINNT\system32\MSTask.exe
TCP msiexec16.exe	952	0.0.0.0	555			LISTEN	TCP	c:\winnt\system32\msiexec16.exe

The admin noted that there was no anti-virus listening at all, but was also mystified by the msiexec16.exe listed as listening on port TCP 555. He did not recognize this process at all. Not to mention the odd port it was listening on for that matter. With this odd fact in mind the admin told the professor to not use his computer until further notice. The admin returned to the main system administration office.

Once there he asked his co-workers if they had heard or knew of this msiexec16.exe process. None of them knew so he signed onto his computer and decided to give Google a chance of finding out what it was. The system administrator was rather alarmed when he saw the results of his google search. He quickly browsed the first link and was now pretty certain that the professor's computer was infected with a trojan.

He quickly related this information to his supervisor. At this time the supervisor decided the system administrator was probably correct in his assumption of a trojan infection. He ordered the admin to go and take a ghost image of the hard drive for possible future evidence purposes. The supervisor was not at all amused that someone had possibly infected the computer with a trojan and wanted to keep some type of evidence. Bearing this in mind the admin was dispatched to do just that. We were now roughly two hours after John had infected the professor's computer.

The college was quite lucky to have caught this breach of computer security so quickly. It was largely due to the fact that the professor was diligent in applying basic computer security principles he had been taught by the in-house computer staff.

Upon the sys admin's return with a ghost image of the professor's hard drive the supervisor got the system administrators together for a meeting. With all of the computer staff largely at the meeting the supervisor now commenced asking people what they thought was going on. All of them with the google search related information in mind agreed it was a trojan. These types of incidents were not something that occurred that often here at the college. Most all of this type of malware was caught by the anti-virus solution.

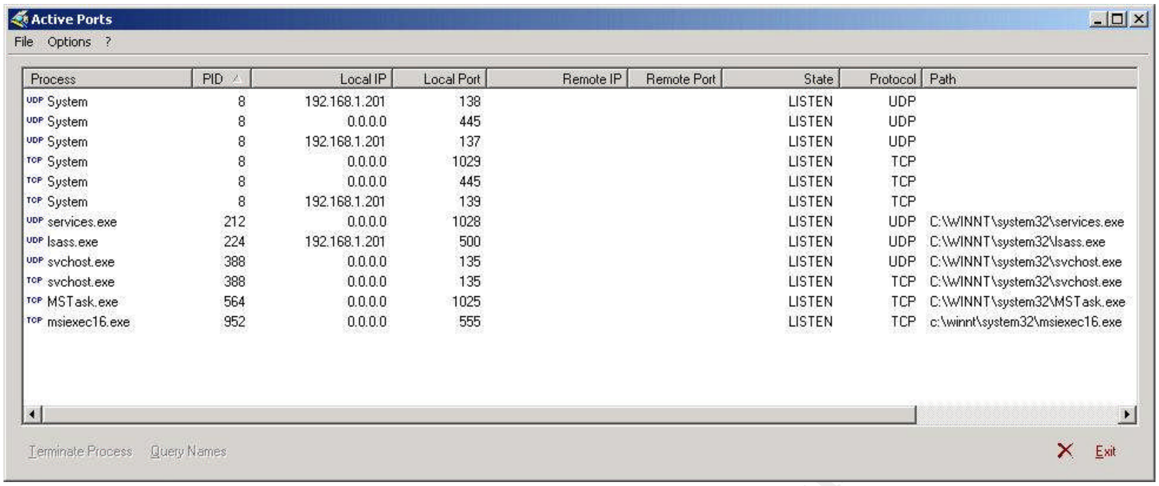
In light of this is why the supervisor was having a meeting to try, and figure out what had happened to make it sure it did not happen again. All of the suggestions on how to fix the problem, or deal with seemed a little chaotic to the supervisor. One of his admins though told him that he know someone in computer security who was a consultant. He could phone his friend and ask him perhaps?

This seemed like a good idea, but the supervisor wanted to know what this friend would charge if anything at all to help them. Within a few minutes the admin reported back that the friend could indeed help out and at what the supervisor considered a reasonable fee. It was now roughly four hours after John had infected the professor's computer. That consultant was me. At this time I advised the supervisor to have someone monitor the professor's computer as it indeed sounded like it was a trojan infection. Lastly I informed the supervisor of two things. Did they want to try and catch who it was that had infected the server or simply clean up the damage as it were? I was told they would like to catch if at all possible who had done this.

One thing the supervisor did not understand was how the college's mail server security software had not stripped off the attachment that was probably used to infect the computer. It also occurred to the supervisor that someone may have also just infected it physically ie: had physical access to it. He knew that most professors had an open door policy for their offices. This was a policy that he had never cared for but campus administration allowed them to do it in an effort to foster a student/teacher relationship.

With my advice in hand the supervisor dispatched an admin to monitor the professor's computer for the remainder of the day. I would be at the college myself physically by the next morning so the supervisor left everything intact on the professor's computer. This was done in an effort to hopefully see someone actively connect to the trojan server on the computer of the professor. The day ended with no connection attempts, and this would be allowed to run one more day before the computer was to be reformatted. Little did John know of what was transpiring. The supervisor was happy at least that he had gotten an image of the professor's computer as evidence if need be.

The next morning (the day after the actual infection) I arrived at the college and spoke with the supervisor. It was decided that I was to confirm if this was a trojan infection and do what I could by days end to identify if possible who had done this. I went down to the professor's office a little prior to his teaching his first class. He related to me what had happened, and then had to leave for class. It was fortunate I noted that Activeports was installed as I very much liked this program. It listed all listening processes and fully connected sockets right to where the program was also installed. This is seen on the screen shot noted below.



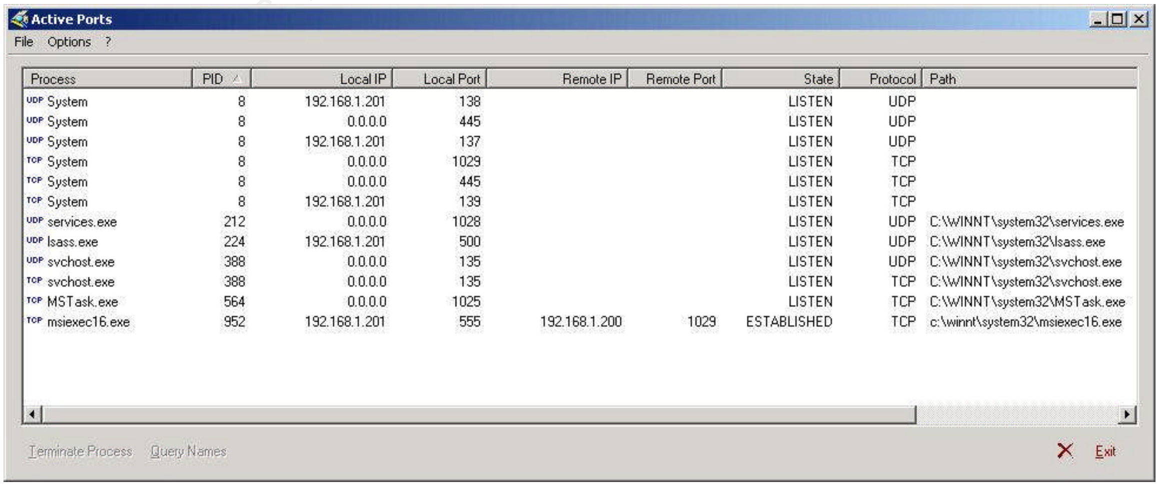
Process	PID	Local IP	Local Port	Remote IP	Remote Port	State	Protocol	Path
System	8	192.168.1.201	138			LISTEN	UDP	
System	8	0.0.0.0	445			LISTEN	UDP	
System	8	192.168.1.201	137			LISTEN	UDP	
System	8	0.0.0.0	1029			LISTEN	TCP	
System	8	0.0.0.0	445			LISTEN	TCP	
System	8	192.168.1.201	139			LISTEN	TCP	
services.exe	212	0.0.0.0	1028			LISTEN	UDP	C:\WINNT\system32\services.exe
lsass.exe	224	192.168.1.201	500			LISTEN	UDP	C:\WINNT\system32\lsass.exe
svchost.exe	388	0.0.0.0	135			LISTEN	UDP	C:\WINNT\system32\svchost.exe
svchost.exe	388	0.0.0.0	135			LISTEN	TCP	C:\WINNT\system32\svchost.exe
MSTask.exe	564	0.0.0.0	1025			LISTEN	TCP	C:\WINNT\system32\MSTask.exe
msiexec16.exe	952	0.0.0.0	555			LISTEN	TCP	c:\winnt\system32\msiexec16.exe

On the right hand side is what is listening and where it is on the hard drive of that computer. The professor's computer in this case. We can see that the professor's IP address is 192.168.1.201 as seen in the screen shot above. I was now able to see that indeed there was "msiexec16.exe" listening on TCP port 555. I also noted that this was in the system32 folder. A favorite place for trojans to install themselves. I decided to install a packet sniffer in the form of windump as this was a Windows platform, and the appropriate libpcap for it as well. With this in place I decided to sniff the traffic coming to the professor's computer. I used the following bfp filter to get the traffic that I wanted. Seen as this trojan was using TCP as its transport protocol I only wanted to see TCP based protocol packets. The filter was built as follows;

windump.exe –w filename –nXvSs 0 tcp and host 192.168.1.201

This would give me a lot of traffic but at least only TCP based traffic. I also decided to log this in binary mode as it would give me flexibility to parse it later.

It was at this time that I noticed while I was setting up that an active connection was established to the professor's computer. Please see screenshot below.



Process	PID	Local IP	Local Port	Remote IP	Remote Port	State	Protocol	Path
System	8	192.168.1.201	138			LISTEN	UDP	
System	8	0.0.0.0	445			LISTEN	UDP	
System	8	192.168.1.201	137			LISTEN	UDP	
System	8	0.0.0.0	1029			LISTEN	TCP	
System	8	0.0.0.0	445			LISTEN	TCP	
System	8	192.168.1.201	139			LISTEN	TCP	
services.exe	212	0.0.0.0	1028			LISTEN	UDP	C:\WINNT\system32\services.exe
lsass.exe	224	192.168.1.201	500			LISTEN	UDP	C:\WINNT\system32\lsass.exe
svchost.exe	388	0.0.0.0	135			LISTEN	UDP	C:\WINNT\system32\svchost.exe
svchost.exe	388	0.0.0.0	135			LISTEN	TCP	C:\WINNT\system32\svchost.exe
MSTask.exe	564	0.0.0.0	1025			LISTEN	TCP	C:\WINNT\system32\MSTask.exe
msiexec16.exe	952	192.168.1.201	555	192.168.1.200	1029	ESTABLISHED	TCP	c:\winnt\system32\msiexec16.exe

This was interesting indeed. Someone within the college itself had just connected to the computer. I had also logged the packets thankfully as I had just installed the packet sniffer. See packet tbelow (only the packet of interest is shown for brevities sake)

```
10:37:47.822736 IP (tos 0x0, ttl 128, id 104, len 85) 192.168.1.201.555 >
192.168.1.200.1028: P [tcp sum ok]
1525306629:1525306674(45) ack 3840986617 win 65524 (DF)
0x0000          4500 0055 0068 4000 8006 7559 c0a8 01c9   E..U.h@...uY....
0x0010          c0a8 01c8 022b 0404 5aea 5505 e4f0 cdf9   .....+..Z.U.....
0x0020          5018 fff4 0165 0000 3030 31ac 4f70 7469   P....e..001.Opti
0x0030          7820 5072 6f20 7631 2e33 3220 436f 6e6e    x.Pro.v1.32.Conn
0x0040          6563 7465 6420 5375 6363 6573 7366 756c   ected.Successful
0x0050          6c79 210d 0a                                ly!..
```

I now knew for certain that this was Optix Pro in use as evidenced by the packet above. I had replayed this packet capture in another DOS prompt when I saw the connection established and saw the packet above. With this event in progress I called the supervisor right away and informed him of what was happening.

The supervisor was quite ecstatic to hear of this now active connection, and even more so to hear it was from the college itself. He hurried down to see me at this point. Upon looking at Activeports he noticed that the IP address was actually statically assigned to one of the student desks in the professor's class!

With this in mind the supervisor was curious as to what to do. I told him there was little point in trying to involve the police as they would probably not send an officer. I told him due to the fact they now could find out who it was this was probably best dealt with immediately. With the dean of students now called and present we all went into the professor's class.

John was meanwhile happily going through the professor's computer and having a grand old time. He did not need to be there as he already had the math exam, but he found it to be fun exploiting the professor's computer. Well as he noticed all of us enter the room and recognized the faces he suddenly had a sinking feeling in his stomach. He quickly killed the Optix client. Too little too late for him though.

The supervisor quickly looked to the desk with the IP address in question and saw John with a rather scared look on his face. He was now pretty sure he indeed had their suspect. We called over the professor to let him know quickly what had happened and had the student dean bring over John to us laptop and all. John was looking distinctly pale and nervous.

Containment

The time was now roughly 24 hours after John had infected the professor's computer with the Optix Pro trojan. All of us were now sitting in the supervisors office. That was the student dean, myself, John, the supervisor and now another member of the faculty for an impartial witness to the proceedings.

I had advised them to get someone else from the faculty to act as an impartial observer. This was someone who up until now had no knowledge of the events and would see them unfold. Both the supervisor and I related to John what we had found on the professor's computer trojan wise and John's own active connection to it. It turns out that John was not long in owning up to what he had done. He admitted to infecting the professor's computer during class time. He had simply slipped out of the class and using the open door policy to his advantage just infected it himself bypassing network perimeter security.

The supervisor informed John that about his only chance of remaining at this college as a student was a full and complete confession. He asked John what other computers he had infected and was told promptly none. He only infected the professor's as he did not want to have to study for the upcoming exam. The student at this time was dismissed and told to go home and report back tomorrow to the Dean's office.

I informed the supervisor that the Optix Pro trojan did not actively try to infect other hosts itself but only actively opened a socket on the computer itself. There was as such no worry of a virus or worm like spreading effect. It would be a relatively simple matter of removing the registry keys that Optix Pro had added and also deleting the actual trojan server. Then have the now active anti-virus solution do a total system scan.

Personally I told him I would prefer to simply reinstall everything, but if there were no backups of critical files already done it was up to them. Once a machine has somehow been compromised I am loath to simply do spot cleaning on it. I much prefer a system format and reinstall. The supervisor agreed and the professor said that nothing on this computer was critical to him.

I had my “jump kit” with me but I had not really used any of it. An image of the computer had already been obtained by the in-house admin staff thereby negating my having to use my USB hard drive. A copy of the program I like to use Activeports had also already been installed as well. I really did not need to use anything I kept on my 512MB USB stick either. I kept a variety of programs on this stick to help me look at a system. I also had a copy of Knoppix STD in my jump kit.

The admin staff though had used a copy of Norton Ghost to image the professor's hard drive. This program was the one they used for ghosting images of the baseline they offered to students. It was used by the staff for the simple reason that it was simple to use. They simply followed the prompts to ghost a hard drive, or in turn load an image onto a computer. The only problem with using Norton Ghost in this way was that it took quite a few cd's too back up the hard drive. Not an overly efficient means to ghost hard drives which is why they were presently looking at better solutions such as an enterprise version.

With John’s confession in hand and knowing how Optix Pro and other such trojans behaved I was fairly confident in telling the supervisor that the trojan was in all likelihood localized to the professor’s hard drive. I was going to make further suggestions to him after I had finished up cleaning the computer so that the sys admin staff could see what I was going to do. It was already decided that the drive was going to be reformatted but we were going to use this as an exercise in eradication as well.

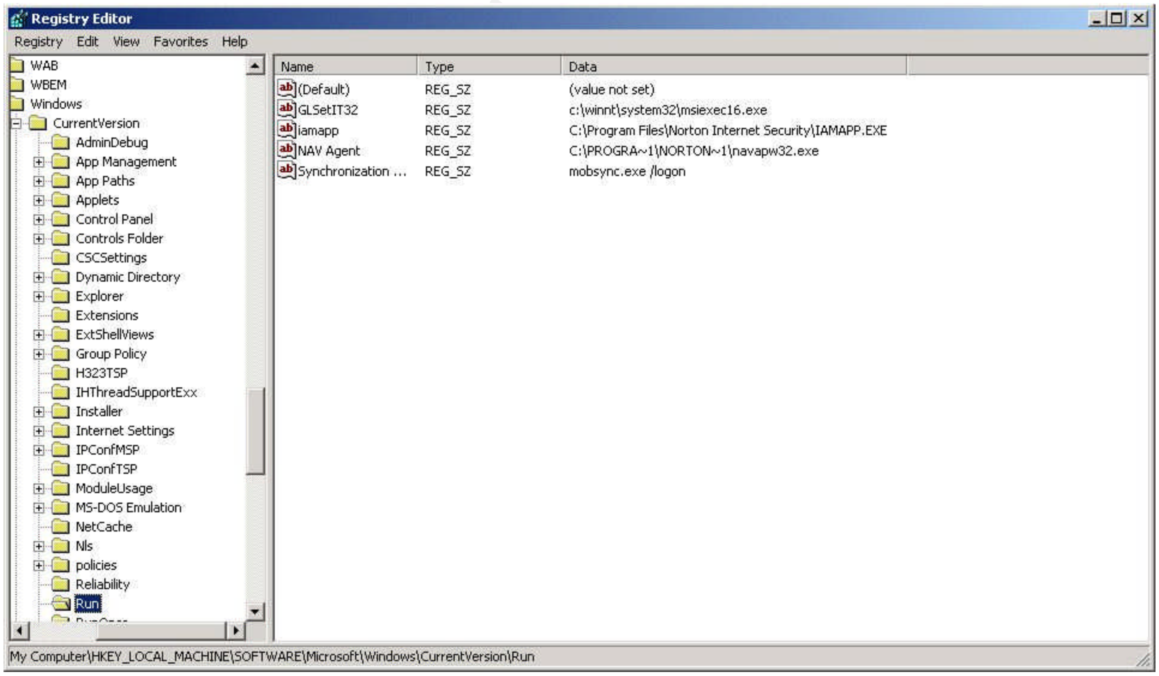
Eradication

I consulted several web sites about Optix Pro and how it works. They all listed the same places that Optix Pro makes registry changes and where it also installs itself. With this information in hand I went about deleting the registry changes and install of the Optix Pro server itself. This was also supplanted by the information that John gave us. He told where exactly he had installed the server as well. This made eradicating the trojan that much easier.

Once at the professor’s computer I went to the known registry key area’s that Optix writes itself too and deleted the entries. The following two registry paths were accessed;

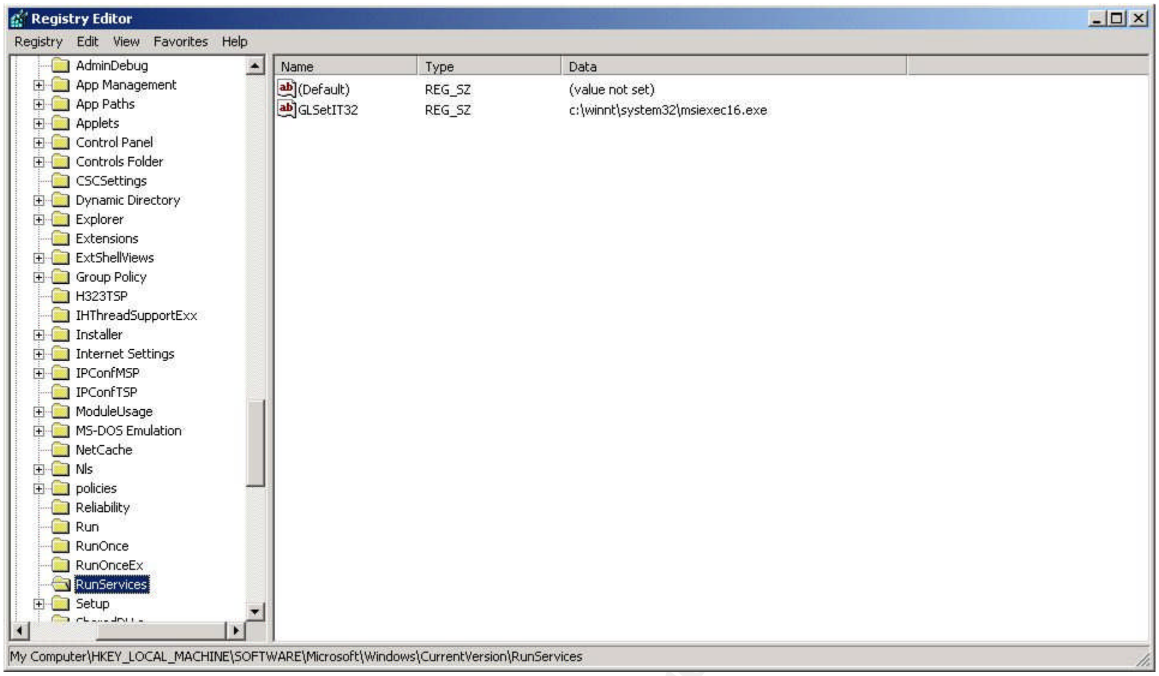
```
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\RunService
s
```

As seen in the below noted screenshots the Optix Pro server disguised as “msiexec16.exe” was indeed there.



Name	Type	Data
(Default)	REG_SZ	(value not set)
GLSetIT32	REG_SZ	c:\winnt\system32\msiexec16.exe
iamapp	REG_SZ	C:\Program Files\Norton Internet Security\IAMAPP.EXE
NAV Agent	REG_SZ	C:\PROGRA~1\NORTON~1\navapw32.exe
Synchronization ...	REG_SZ	mobsync.exe /logon

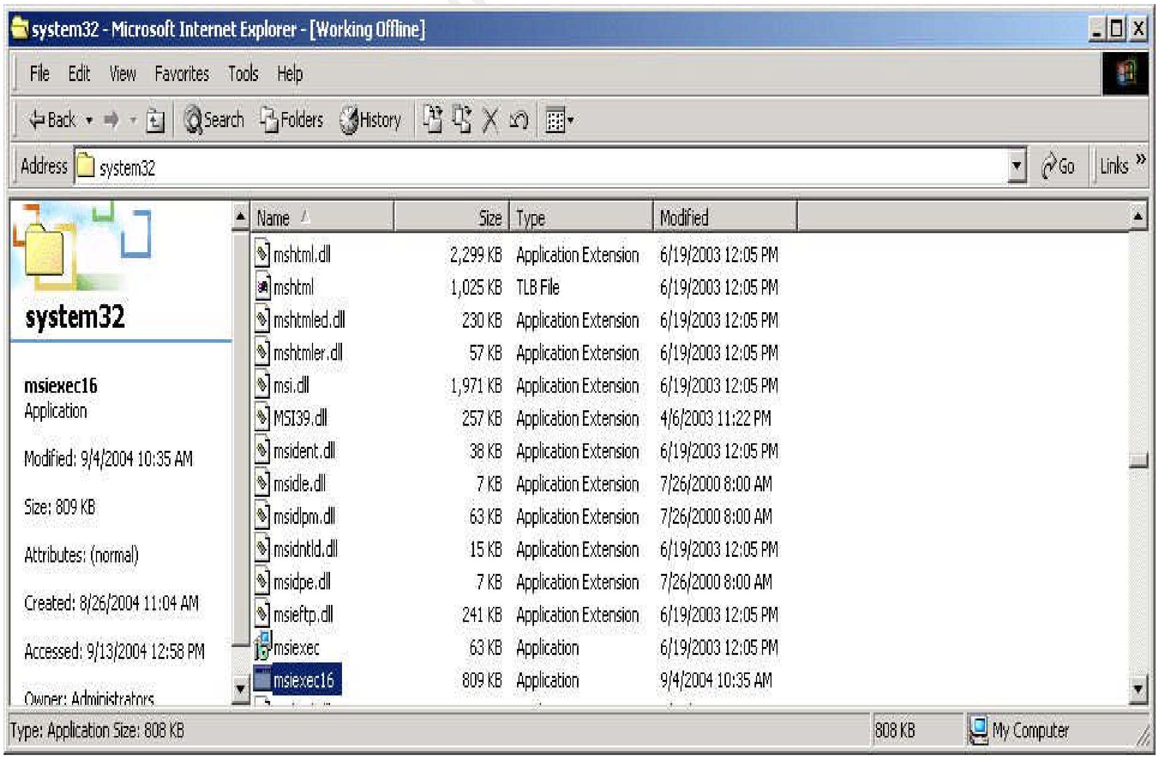
My Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run



Name	Type	Data
(Default)	REG_SZ	(value not set)
GLSetIT32	REG_SZ	c:\winnt\system32\msiexec16.exe

My Computer\HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\RunServices

These two above noted registry changes made by Optix Pro were deleted by myself. I now had to delete the actual executable on the professor's computer ie: the actual Optix Pro server. This normally installs to the following path on Windows 2000 machines; C:\Winnt\System32 I deleted the entry of "msiexec16.exe" that I found there as well. Please note the screen shot below.


system32 - Microsoft Internet Explorer - [Working Offline]
File Edit View Favorites Tools Help
Back
Search
Folders
History
Address
system32
Go
Links
system32
My Computer

msiexec16
Application

Modified: 9/4/2004 10:35 AM

Size: 809 KB

Attributes: (normal)

Created: 8/26/2004 11:04 AM

Accessed: 9/13/2004 12:58 PM

Owner: Administrators

Name	Size	Type	Modified
mshtml.dll	2,299 KB	Application Extension	6/19/2003 12:05 PM
mshtml	1,025 KB	TLB File	6/19/2003 12:05 PM
mshtmled.dll	230 KB	Application Extension	6/19/2003 12:05 PM
mshtmler.dll	57 KB	Application Extension	6/19/2003 12:05 PM
msi.dll	1,971 KB	Application Extension	6/19/2003 12:05 PM
MSI39.dll	257 KB	Application Extension	4/6/2003 11:22 PM
msident.dll	38 KB	Application Extension	6/19/2003 12:05 PM
msidle.dll	7 KB	Application Extension	7/26/2000 8:00 AM
msidlpm.dll	63 KB	Application Extension	7/26/2000 8:00 AM
msidntld.dll	15 KB	Application Extension	6/19/2003 12:05 PM
msidpe.dll	7 KB	Application Extension	7/26/2000 8:00 AM
msieftp.dll	241 KB	Application Extension	6/19/2003 12:05 PM
msiexec	63 KB	Application	6/19/2003 12:05 PM
msiexec16	809 KB	Application	9/4/2004 10:35 AM

Type: Application Size: 808 KB 808 KB

With these registry keys, and trojan server deleted I rebooted the computer. The anti-virus was now up and running as it should be. I initiated a full system scan and once it was done the anti-virus program found no trace of the Optix Pro trojan. The computer now was cleansed and the trojan eradicated.

The trojan itself was easily removed from the afflicted system. It was a well known trojan with well documented analysis from anti-virus vendors. That plus the fact that our anti-virus solution is known to detect this trojan helped as well. I as mentioned rescanned the computer once I had eradicated it's presence on the computer, and no traces were found by the anti-virus program.

This plus the fact that the student John was helpful in detailing what he had done with the trojan on the professor's computer was helpful. I suggested to the supervisor of the system administration department that the computer should still be formatted. Once a computer has been compromised it is always a worry that something may have been missed. At this point the supervisor agreed and with the professor not objecting the computer was then reformatted. Once done the image was reinstalled on the computer. It was lucky that the professor had no critical files on his computer as this facilitated the option of a simple reformat.

The overall root of the problem though lied in the fact that the college had an open door policy during school hours. Had the professor locked his door while he was teaching class, or otherwise while he was not in his office none of this would have happened. That or John the student would have had to look for another way to infect the professor's computer.

Recovery

The steps detailed above in the eradication portion apply to this section as well. All of the trojans registry changes and the actual trojan server themselves were deleted by me. This was also further checked with online analysis of this trojan by anti-virus vendors. Finally the system itself was given a full system scan once the anti-virus was rendered operable by the removal of the trojan itself. The full disk scan found no further traces of the trojan and nor were there any more listening sockets on the computer on TCP port 555.

This cleanup was relatively simple as it did not entail a buffer overflow of a service and the attacker possibly transferring over a rootkit or survey kit. It was helpful as well to have the perpetrator cooperating in the removal of it by telling us where he had installed it.

Through the full anti-virus scan and a checking of active sockets I was confident that the trojan had successfully been eradicated. This in addition to the registry keys and server itself being deleted by myself. With the afore-mentioned tests complete I was confident once again that the trojan was gone. If the registry keys were gone and the server itself as well there was nothing to open a socket on the computer to listen for connections.

Though, as I mentioned I did not entirely trust the student John. He could have transferred any other type of malware over there. This is why I suggested to the supervisor that the system simply be reformatted. Once compromised I find it very difficult to ever fully trust that configuration again.

The system administration supervisor now had ample reason to revisit the issue of having an open door policy with the college's administration. He would do so that very same day and illustrate the dangers of having such a policy in place. With this computer breach just 1 day old he was confident the administration would agree a change of policy was in order.

Lessons Learned

The whole trojan infection incident revolved around one key piece of information or policy. That being that the college had an open door policy for the professor's in which they had literally their offices open during school hours. It was done to help foster a student/professor relationship. In this case though it resulted in a breach of computer security; specifically a breach of physical security.

Physical security is an area quite often neglected I told the supervisor. He readily agreed and was never fond of the policy that the college administration had in place. Luckily the supervisor realized and mostly due to the professor himself this breach was quickly contained. This incident would allow the supervisor to revisit this open door policy with the college administration, and hopefully make they see the dangers of it.

Though I told the supervisor that realistically for this college environment there should be far more security in place. I related to him that placing some intrusion detection systems throughout the college network is pretty much a must. Had the college had some there would almost definitely been an alert triggered by this exchange of trojan client to trojan server.

Also I suggested to the supervisor that they really should go with a centralized anti-virus solution. I mentioned a solution that would automatically push updates to the client computers, and not allow the anti-virus process itself to be terminated. Centralized control is most helpful in managing and consolidating this anti-virus protection.

It was also suggested by me to the supervisor during my meeting with him and the other admins who took part of the original meeting that there really should more then a software baseline. The baseline installed should also have downloads disabled in the Internet Explorer browser, as well as require admin privileges to install any programs. This would force the user to only use what was on the baseline image and not be able to install, or potentially download malware.

Once the supervisor, admins, and myself had discussed the above noted we summed up with some thoughts;

1) Restrict physical access
2) Centralize anti-virus solution
3) Install intrusion detection systems
4) Redo the software baseline with tighter restrictions on it
5) Form an incident handling team to deal with future incidents
6) Obtain formal training in Incident Handling and computer security

With these steps in place the college network would be far more secure, and a response team in place to handle future incidents. It was also key, as well to help maintain the system administrator's skill sets. This would help them deal with any future computer security issues.

Extra's

I have included within this paper such things as packet traces to clearly illustrate certain points, as protocol usage among others. Beautifully illustrated by this link is the reason why you should not use malware, or at least read the source code before you do ☺ Also see a full list of Optix Pro v1.3 capabilities below.

```
2.FEATURE LIST

v1.33 - Client Side
COMPATIVBLE WITH ALL PAST SERVER VERSIONS! in a limited way! (own risk)
Client SOCKS 4/5 Support
Power Options - logoff,suspend,reboot,shutdown etc.
Server Information - Get info about builder settings
File Manager
Process Manager
Windows Manager
Registry Manager
FTP Manager
SOCKS 4/5 Server
Remote IP Scanner
Port Redirect
Application Redirect
Service Manager
Message Box
Matrix Chat (Client-2-vic)
Client-2-Client chat
Computer Information
Get Passwords - (RAS/Cached - 9x and AIM)
Online Key Logger - (now window titles)
Screen Capture with left click mouse manipulation
Keyboard Manipulation - (more advanced)
Cam Capture
SendKeys - old version of SendKeys for older servers
Humor normals - Flash keyboard lights, Monitor on/off, Disable
keyboard/mouse etc.
Humor Screen Printer - print text to their screen!

v1.33 - Server Side
```

```
COMPATIBLE WITH ALL PREVIOUS CLIENT VERSIONS! in a limited way! (own
risk)
Configurable:
Notification Information Separators
IP Address Separator
Info included in any Notification
Idenfitication Name
Server Port
Server Password
Fake Error
Server Icon
Registry Run startup
Registry RunServices startup
win.ini startup
system.ini startup
s7 special method startup!
Server File Name
Start Directory (windir/sysdir)
Melt Server
Unlimited ICQ Number Notification
Unlimited CGI Script Notification
Unlimited IRC Server/channel Notification
Unlimited PHP Script Notification
Unlimited SMTP Notification
Toggling killing of in-built exe/service list for firewalls
Toggling killing of in-built exe/service list for Anti-Virus
Toggling killing of in-built exe/service list for packages classifed as
both anti-virus and firewall!
Unlimited Number of custom exe's to kill
Unlimited Number of custom services to kill
Easily Automated UPX Packing if needed.
Option for unpacked or packed server with your own packer if wanted
(instructions clear)
```

This above noted list came from the following site

References

1) http://www.evileyesoftware.com/ees/content.php?content.8
2) http://www.securiteam.com/securityreviews/2RUQ1RPRPA.html
3) http://www.security-forums.com/forum/viewtopic.php?t=9359&start=0
4) http://www.diamondcs.com.au/index.php?page=archive&id=analysis-optixpro
5) http://securityresponse.symantec.com/avcenter/venc/data/backdoor.optixpro.13.html
6) http://www.giac.org/practical/GCIH/Sherman_Hung_GCIH.pdf