Malware and anti-malware (fun with Trojans)

License

Contents

General notes about the labs

Many of the tasks you complete within our labs could be considered illegal if targeted at a computer that you do not have explicit permission to interact with, do security tests on, and attack. In short, keep all activity contained to our labs and to computers you have legal permission to attack. Use common sense, and act within the law, ethically, and according to your own morals. With power comes responsibility, use it wisely.

Often the lab instructions are intentionally open ended, and you will have to figure some things out for yourselves. This module is designed to be challenging, as well as fun!

However, we aim to provide a well planned and fluent experience. If you notice any mistakes in the lab instructions or you feel some important information is missing, please feel free to add a comment to the document by highlighting the text and click the comment icon (), and we will try to address any issues. Note that your comments are public.

The labs are written to be informative and, in order to aid clarity, instructions that you should actually execute are generally written in this colour. Note that all lab content is assessable for the module, but the colour coding may help you skip to the "next thing to *do*", but make sure you dedicate time to read and understand everything. Coloured instructions in *italics* indicates you need to change the instructions based on your environment: for example, using your own IP address.

You should maintain a lab logbook / document, which should include your answers to the questions posed throughout the labs (in this colour). You do not need to submit your lab book for assessment, but it is designed to be useful for your own study and for preparation for the final test.

Preparation

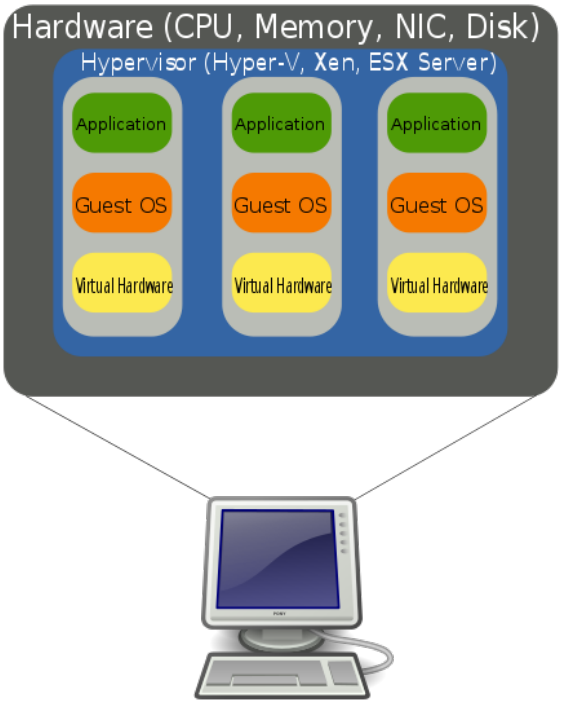
Start by loading the latest version of the LinuxZ template from the IMS system. If you have access to this lab sheet, you can read ahead while you wait for the image to load.

> To load the image: press F12 during startup (on the boot screen) to access the IMS system, then login to IMS using your university password. Load the template image: LinuxZ (load the latest version).

Once your LinuxZ image has loaded, log in using the username **student** and password **theIliad**.

Virtualisation and our use of virtual machines (VMs)

Virtualisation is a very powerful tool, that can provide important security features, such as isolation, and is an important component of many modern cloud infrastructures. Virtualisation can create virtual environments, and can even run entire operating systems as though they were on separate hardware. This type of virtualisation is known as platform virtualisation, or hardware virtualisation. As illustrated in the figure below, virtualisation allows one set of hardware (a computer), to host a number of guest virtual machines (VMs), each with their own operating systems, applications, and virtual hardware.



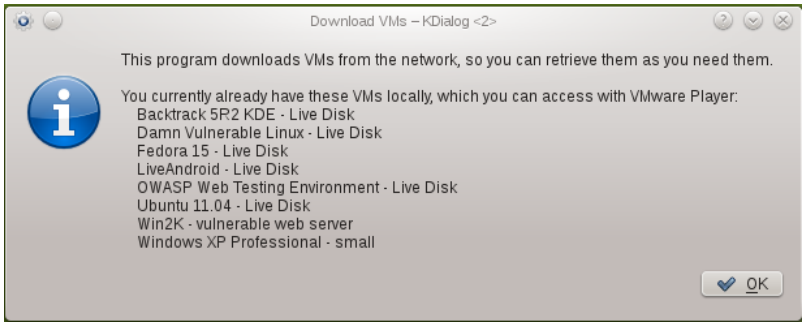
Hardware Virtualization (Image: public domain by John Aplessed)

During the module we will make use of various VMs to recreate various scenarios.

To make it easy to set up the various lab tasks and hacking scenarios, we have a network share containing a number of different VMs, including some systems to launch security attacks, and some to be attack victims.

To download VMs:

On the LinuxZ desktop, click the “download VMs” icon.



VM download script: already installed list

As shown in the figure above, you will be greeted with a list of VMs already on your system. These VMs can be launched from the "launch VMs" desktop icon. Note some VMs run as Live disks off the ISOs on the network share. These are very lightweight, since you do not have to download the VM, it just runs over the network. However, with the Live disk VMs your changes will not persist when restarting the VM:you will lose any changes within the VM. Click "OK".

If you want to download VMs, you can do so at the next step. Simply select which of the VMs you want to download to your local system (you can select multiple at a time).

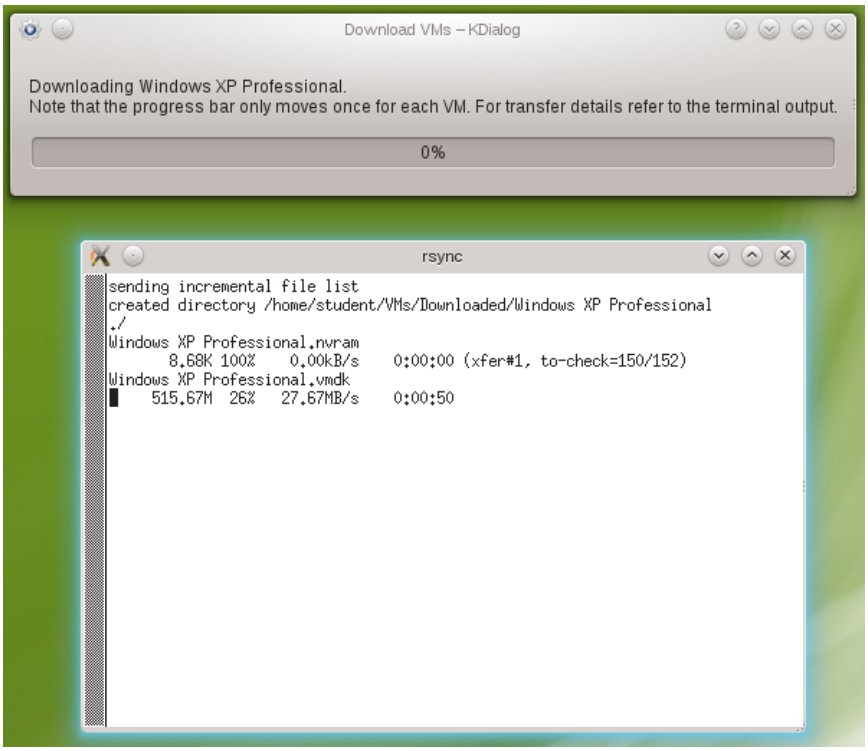
In this case, select the "**WinXP Pro - bridged with network share**" and "**Windows 98 - with VMTools and Trojans**" VMs, which we will use both as our attacking and infected systems.

Why are we using such old versions of Windows?

> We won't be using Windows 98 in any other labs in this module (*relax!*). This is a blast from the past to illustrate the foundations of Trojan horse programs (the same theory without some of the frills). Also, the Windows 98 VM is only 230MB(!), so it is very fast to get going. Windows 98 has almost none of the security features we expect from an operating system these days.
>
> Windows XP contains the initial versions of many security features (for example, it includes firewall software), although it does not have all the security features found in newer versions of Windows. It is a *much* smaller download than Windows 7 or 8. Also, Windows XP is still currently deployed in many corporations and used by many home users and it is therefore a good system to use as an introduction to forensics and security.
>
> Our other forensics and security modules make further use of Linux, Unix, and more recent versions of Windows.

Click "OK" to begin the download. As shown below, you will then be able to view the download progress. While waiting on the download, please read ahead.

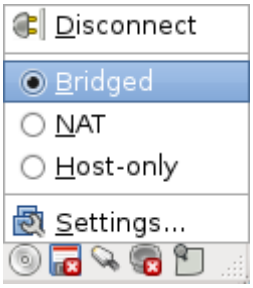
Once complete, you will be asked whether to launch the VMs now. Click “Yes”. If prompted, tell VMware “I copied it”.



VM download script: downloading

Configure the both VMs to use Bridged networking:

After starting the Win98 VM, click in the bottom right network interface icon () and select “Bridged”:



Restart the VM (or in an MSDOS prompt in Win98 run “ipconfig /release_all” then “ipconfig renew_all”).

Introduction to malware

"If a bad guy can persuade you to run his program on your computer, it's not your computer anymore"

Microsoft, "TechNet Essay: 10 Immutable Laws of Security"

This week's lab is a gentle introduction to some important security concepts, such as the damage that can be caused by malware. The lab involves using a real-life Trojan horse to take control of infected computers.

The security issue is that programs do not always act in the best interests of the users that run them. Yet typically when you run a program it can do anything that you have permission to do. Often end users are forced to trust that applications are acting correctly. If the author of the software you are running wants to do malicious things, then the author basically has the power to use your privileges however they please.

An attacker just needs some way of getting malware onto the systems of end users...

Question: Lookup the meanings of the following terms, and record these in your **logbook**:

Malware:

Virus:

Worm:

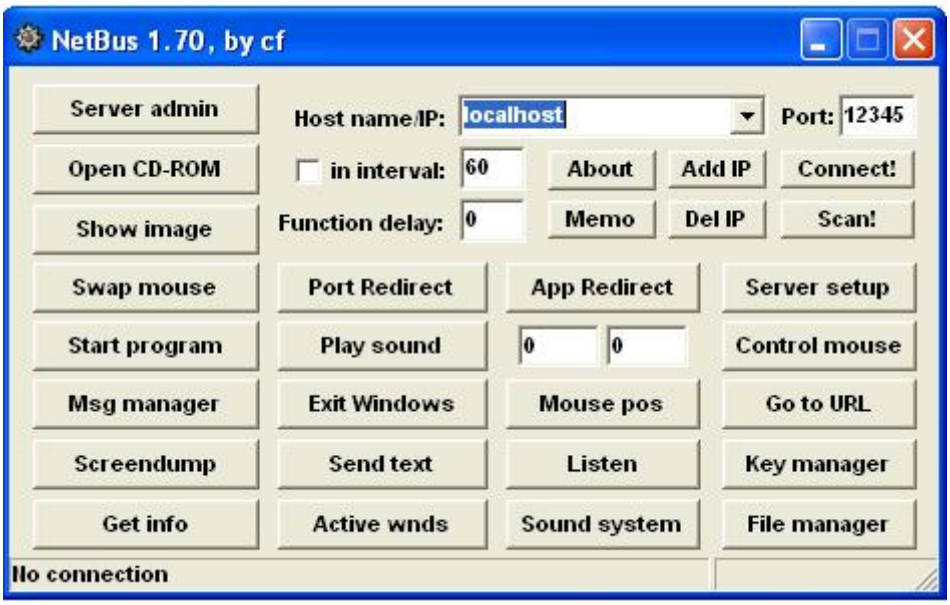
Trojan horse:

Trojan Horses and NetBus

To illustrate the security threat of malware, we will experiment with a RAT known as NetBus.

Remote access Trojans (RATs) are a category of Trojan horses that hide themselves on an infected computer, and enable a malicious attacker to gain remote access to the infected system, and to remotely control the system with features that are similar to legitimate remote network administration services. The attacker can typically instruct the Trojan to do practically anything they wish on the target computer, such as access files or format harddisks.

NetBus was popular with attackers in the 90s, due to its ease of use. If you can convince or trick someone to install the server component, then using the client you can connect to their computer and take control of it: including copying files and listening in to connected microphones. The figure below shows the client interface, which an attacker would use to send commands to an infected computer. Note the range of features available to an attacker, via a simple interface.



NetBus 1.7 Trojan horse client

RATs and client-server models

Like many network services, RATs typically employ a *client-server model*.

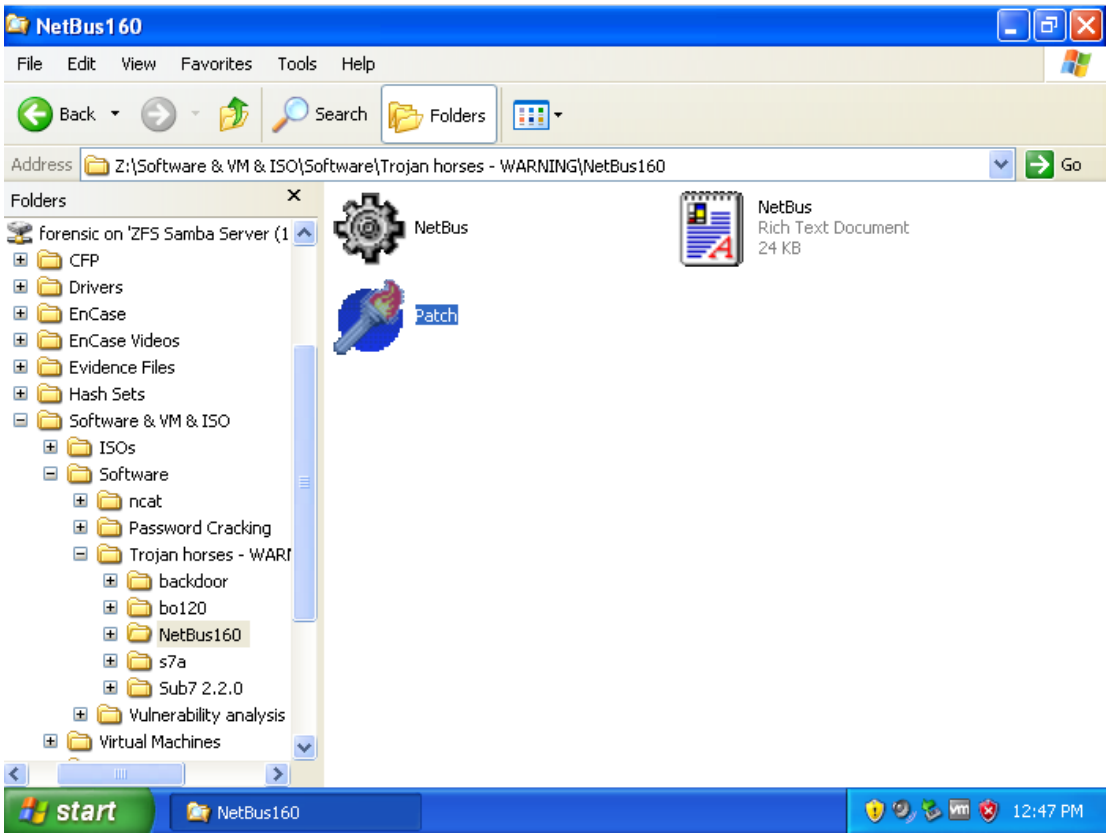
A client-server approach to networking involves clients connecting over a network, such as the Internet, and sending requests to servers. A server typically binds to a network address and awaits a connection from a client. A client initiates a connection with the server and sends a request. The server then processes the request and sends a result to the client. Note that the client-server model is how most Internet protocols work, such as the worldwide Web (for hosting web pages), email, and file transfer protocol (FTP).

Many RATs involve a server program (which could be considered the Trojan horse itself), that when run on a system hides itself and gains persistence on the system, so that the malicious service starts when the system boots. An attacker's first task is to devise a way of getting the server program to run on the victim system.

Looking at the NetBus server program, and file type extensions

From within the **Windows XP VM**,

> Open Windows Explorer (WindowsKey+E), and browse to the directory (folder) containing NetBus. NetBus is located in:
>
> > "Z:\Software & VM & ISO\Software\Trojan horses - WARNING\NetBus160\"



Viewing the network drive from within the VM

Note that this directory contains a file "Patch.exe", which is the executable that will infect a system with the RAT. *(Don't run this at this point!)*

The first aim of an attacker is to get a *victim* to run this program. Perhaps the easiest way to do so is by using some *social engineering*; that is, by manipulating people into performing actions for the attacker.

The most straightforward approach would be to directly ask someone to run the program; perhaps by emailing the file to someone and writing something that would be likely to entice the target into running the program.

Because people with some security awareness are aware that running an ".exe" file can be dangerous, the attacker may choose to disguise the file further.

Open the NetBus.rtf document, and have a quick skim through some of the information.

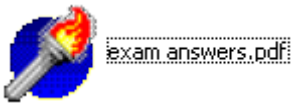
Disguising the file extension: basic rename

Copy "patch.exe" and "NetBus.exe" to your "My Documents" directory.

Windows typically hides file extensions by default, which can make it very easy to hide the true purpose of a file from an unsuspecting victim.

Rename "patch.exe" to "exam answers.pdf.exe".

If you have file extensions hidden, you will see the following, which may be convincing enough to trick some users (although hopefully not many):



Renamed Trojan

Changing the file icon

Although changing the filename may be somewhat convincing, the icon likely makes the file seem suspicious. Therefore the attacker may wish to further disguise the file by changing the icon embedded within the program.

There are many different programs that can be used to edit the resources embedded in executable files, such as icons. One of which is "Resource Editor".

Start Resource Editor:

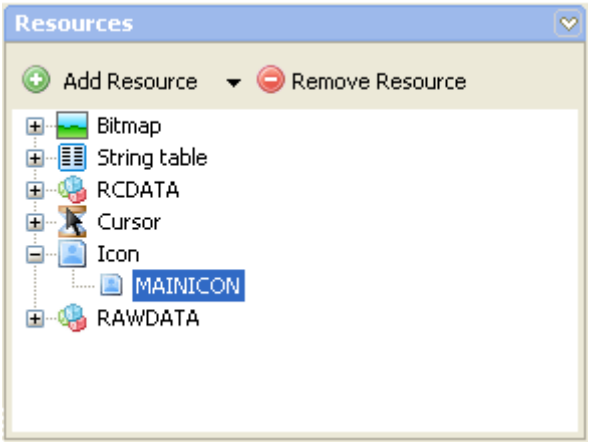
> Copy the Resource Editor program to your My Documents directory. The program can be found in the following network share:
>
> > "Z:\Software & VM & ISO\Software\ResourceEditor20110910\ ResourceEditor.exe"
>
> Run the local copy.

Within Resource Editor, open "exam answers.pdf.exe".

> Tip: browse to My Documents, and select Files of type: All files(*).

Change the icon:

In the “resources” area, select Icon/MAINICON.



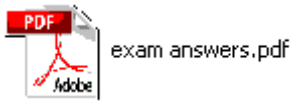
Resource Editor resources

In the Edit (Image) area, click the import item button (), and import an Adobe Reader icon.

There is an example icon in the “Z:\Software & VM & ISO\Software\ResourceEditor20110910” directory, or you can download an icon from the Internet.

Save your changes (Menu: File/Save, or Ctrl-S).

In Windows Explorer, have another look at your Trojan program. It is starting to look more convincing.



Renamed Trojan with a modified icon

Disguising the file extension: screensavers

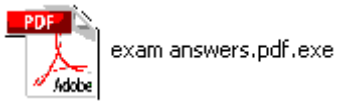
Most advanced users would have file extensions set to visible.

Set the folder properties to show file extensions:

In Windows Explorer, open “Tools” (menu), “Folder options…”.

Go to the “View” tab, and in “Advanced settings” uncheck “Hide extensions for known file types”.

Our Trojan no longer looks so inconspicuous.



Renamed Trojan with a modified icon showing file extension

There are many file extensions that run executable code on Windows. Extensions such as .com, .cmd, .bat, .lnk, .vb, and .scr also run code directly on your system. Creating a screensaver executable can be as simple as renaming an .exe file to have a .scr extension.

Rename “exam answers.pdf.exe” to “exam answers.pdf.scr”.

Autorun

Yet another trick is to put your enticing file on a DVD-R or USB drive, and leave it somewhere it is likely to be found.

To make matters even worse, some systems (including older Windows systems, before Windows 7) will automatically launch software when a disk is inserted.

If you have a USB handy you may want to try this:

> Simply put a copy of the Trojan on the root (main directory) of a USB or optical disk (such as CD/DVD), rename the file to avoid spaces, then in Notepad enter:
>
> ```
> [autorun]
>
> open=examanswers.pdf.scr
> ```
>
> Save the file as “Autorun.inf” in the root directory of the disk.
>
> Insert your disk into the Win98 VM, and the system should automatically be infected with the NetBus server.

Disguising the file extension: using the “Unitrix Exploit”

Many versions of Windows also suffer from a file naming attack known as the “Unitrix” exploit. It is named after the Unitrix malware which made use of this trick. By including the Unicode character “U+202E” the filename is displayed incorrectly, even when file extensions are set to be displayed.

Unicode enables characters from most languages to be represented, such as “叶”, “葉”, “あ”, and “말”, but can also represent non-visible control codes such as the instruction to display text from right-to-left. This can have the effect of displaying the actual file extension within the middle of the file name.

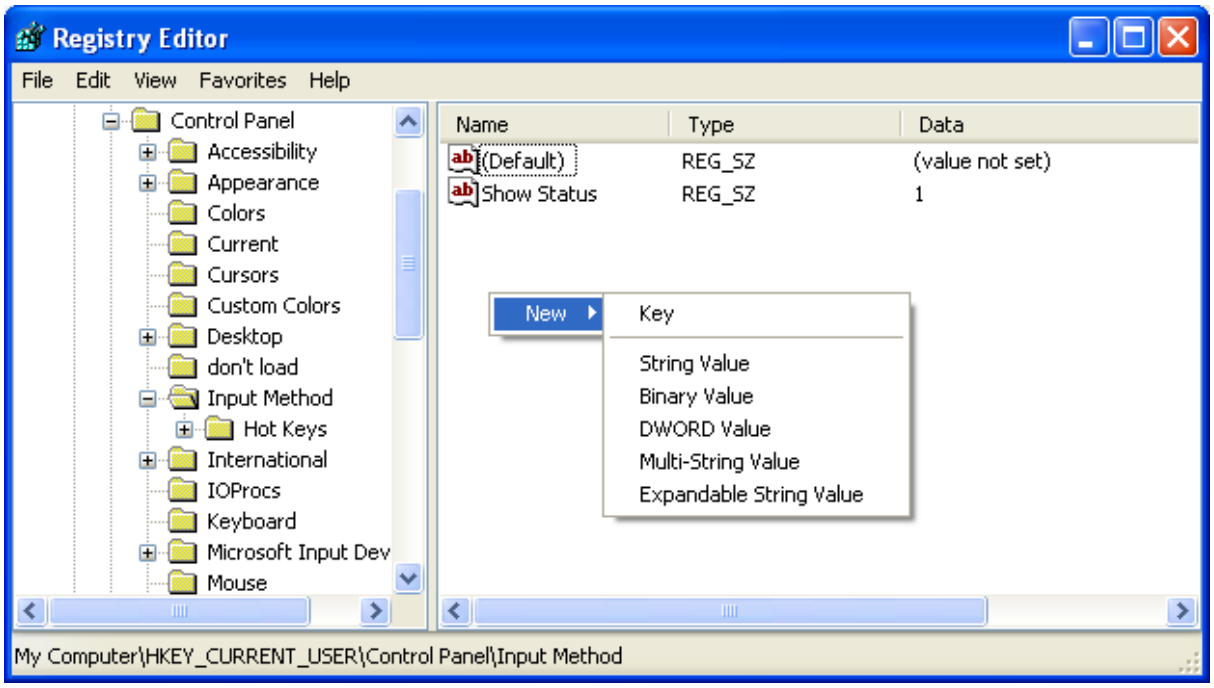
In order make this name change, we need to edit the registry:

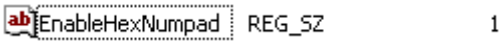
Open the registry editor (WindowsKey+R, “regedit”).

Browse to:

“HKEY_CURRENT_USER\Control Panel\Input Method”

Create a new String Value named “EnableHexNumpad”, with a value of “1”.



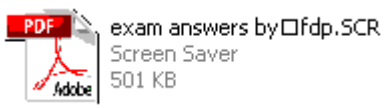


Regedit enabling Hex Alt code input

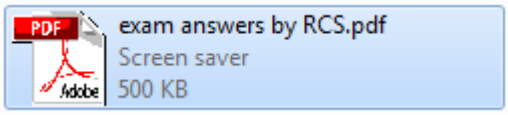
Reboot your Windows XP VM, so that these registry changes take effect.

Press F2 to rename your file, clear the current file name and type “exam answers by ” (Space), then hold down ALT and press “+” on the numpad, and type “202e”, then release the Alt key and type “fdp.SCR”. Press Enter to confirm the rename.

On a Windows system that does not support right-to-left within filenames, it will look like this:



However, on a system that *does* support right-to-left in filenames, such as Windows 7, it will look like this:



LinuxZ openSUSE system is also fooled into displaying a deceptive file extension:

Another common trick is to include a lot of space within a filename, for example:

"funnycats.jpg .exe".

Some programs will display this without the final extension.

Edit the file name to something different to something you think may fool someone into running it, with or without the Ultrix exploit.

Question: What other ways can you think of for getting a Trojan horse program running on a target system?

(Ethically) attacking your classmates!

So now that you have created a crafty filename for your Trojan horse, lets infect someone's system, and experiment with what the Trojan enables us to do.

For the sake of your classmates' enjoyment, infect your own Windows 98 VM by running the NetBus server (within the Windows 98 VM):

> **In the Win98 VM**, a copy of the NetBus Trojan can be found in "C:\Trojans\NB160".
>
> Alternatively, you can copy your renamed program into the VM, which you may have named "exam answers.pdf.scr".
>
> Yes, even though you would be silly to intentionally run a Trojan horse on your own computer at home (outside of an isolated VM), do this step.
>
> If you feel so inclined you could set a password by connecting to your own system using the NetBus client, and share the password with those you are happy to play with your system. However, I encourage you to just have fun

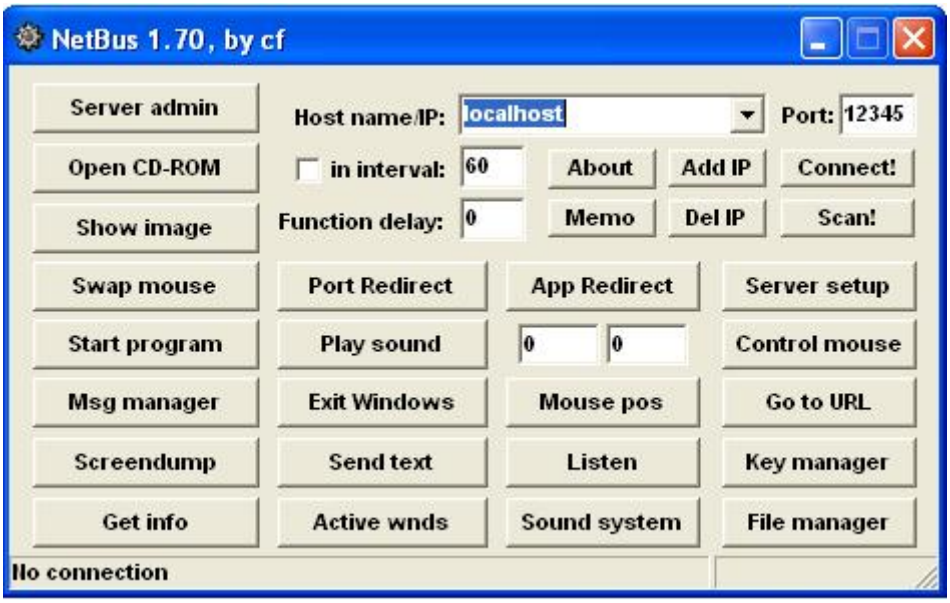
with it, and go in without a password set. (Also, the password is not actually implemented securely, so someone who knows what they are doing can circumvent it.)

On your WinXP VM, start the NetBus client interface (NetBus.exe).

If you have followed these steps, you now have an infected Win98 system, and a clean WinXP system that you can use to control the infected system(s) in the lab.

Find a classmate's Win98 IP address that has done the same, and connect using the client:

> If you are working alone you can alternatively connect to your own Win98 VM, by entering its IP address.
>
> **On your Win98 VM**, find your IP address. Open a command prompt, and run:
>
> ```
> ipconfig
> ```
>
> Share your IP address with those around you.
>
> Run the client program “NetBus.exe”, type their IP address in the Hostname/IP text box, and press “Connect!”.



NetBus 1.7 Trojan horse client

Take the time to explore the many features that the attacker has access to with NetBus.

Try controlling their mouse, record and view what they are typing on their system, upload files to their computer, and so on.

Tip: to find a list of the infected systems on the network, from LinuxZ run: "`nmap --open -p 12345 192.168.202.0/24 | grep 192.*`"

After you have explored NetBus, try installing the NetBus server on your Windows XP VM, and see what does and does not work for others controlling your system.

If prompted, allow the NetBus Trojan through the firewall. When NetBus was first released, Windows did not include a firewall. With Windows XP an attacker would need to need to also trick the target into letting the traffic through the firewall.

Note that Windows XP was released after RATs became popular, yet most of NetBus's features still work in Windows XP.

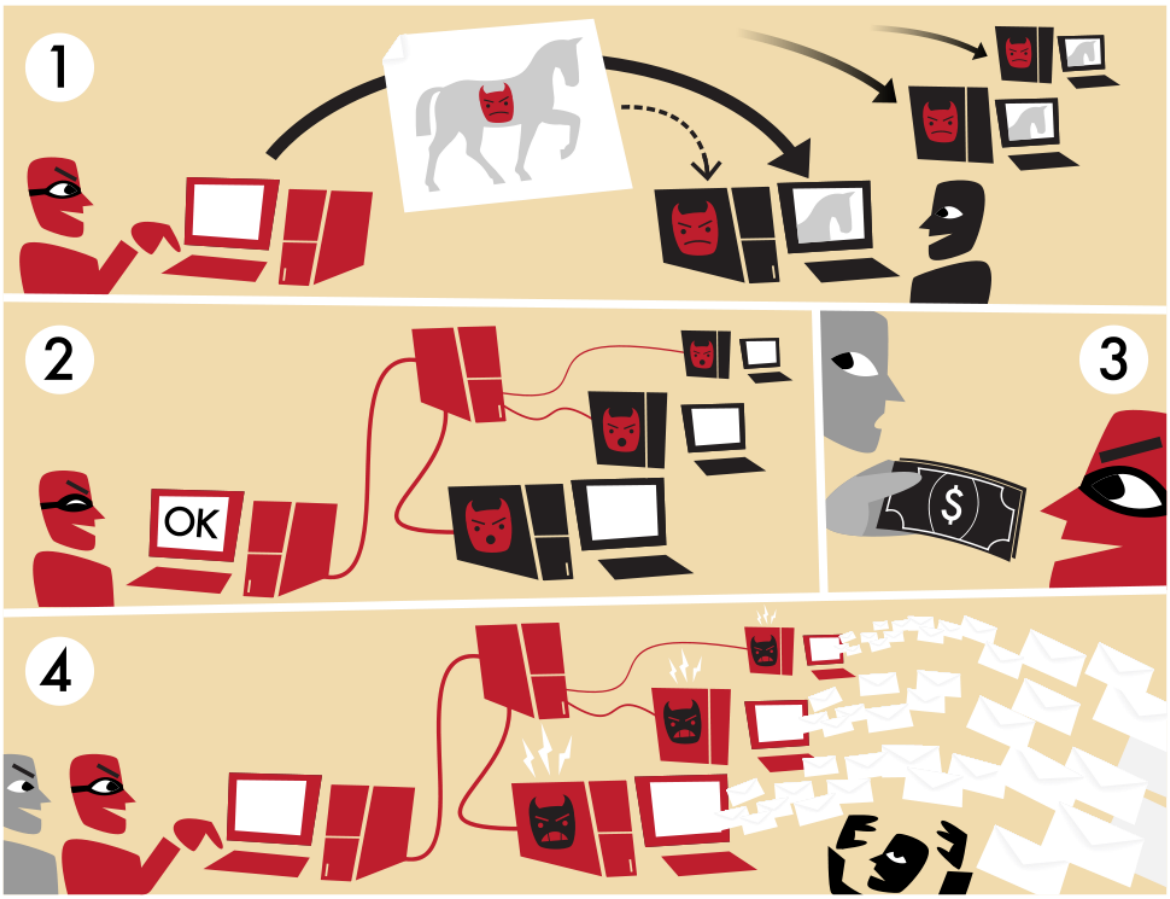
There are a number of other Trojan horse programs available in the Trojan directory (in the Win98 VM and the network drive). Feel free to **explore using these other Trojan horses (servers and clients)**. SubSeven is a good example of a slightly more advanced Trojan, which works best in the Windows 98 VM.

If you like, you can download VMs of various versions of Windows from the network (as explained above), and experiment with NetBus to determine what features do and don't work on older or newer versions of Windows.

Botnets and distributed Trojan networks

A more sophisticated use of Trojan horses has become popular with attackers: botnets. A *botnet* is a network of compromised systems under the control of a botnet operator.

As illustrated in the figure below, in a centralised botnet: 1) the attacker finds a way of getting Trojan programs onto various people's computers, 2) the infected systems connect back to a command-and-control (C&C) server and will wait for instructions from the operator, 3) someone may pay the botnet operator to put the 'bots' to work, 4) the operator sends a command via the C&C server, to instruct the botnet to send spam (or harvest credit card details, and so on).



How a botnet works (Image: by Tom-b licensed under the Creative Commons Attribution-Share Alike 3.0 Unported license)

Newer botnets have become further advanced and rather than taking a client-server, or centralised network model, the most advanced botnets are based on a peer-to-peer network.

A *peer-to-peer* (p2p) network is decentralised and distributed, so that there is no central "weak spot". Other examples of peer-to-peer networks include Spotify's streaming media, Bittorrent file sharing, and the Bitcoin digital currency. P2P can be a useful design architecture.

In P2P botnets, each bot may forward verified requests from a botnet operator to other bots in the network, which means taking down the botnet is more complex than taking down a central C&C server.

Question: What kinds of activities would a Trojan horse or botnet operator gain from? For example, they may be paid by advertisers for sending spam.

Trojan horses and protections

Once you have finished exploring what you can do using these Trojans, install a free antivirus product, such as Microsoft Security Essentials, Comodo Antivirus, or AVG Free.

Comodo Antivirus is available on the network drive, in:

"Z:\Software & VM & ISO\Software\Antimalware\"

Try running the NetBus server again. The malware protection software should **detect and remove the Trojan horse**.

Question: Does having anti-malware software installed protect you against all Trojan horses?

Why not? (Hint: what about new ones?)

Question: Why are Trojan horse programs able to do so much damage?

Conclusion and reflections

Question: Given these risks, when do you think it would be appropriate to download executable programs from the Internet? From which sites?

You have seen that when a computer is infected with a Trojan, it is possible for an remote attacker to take control over a computer.

The "Trojan defence" is the legal defense that a Trojan horse was responsible for charges against the defendant. Basically, the argument is made that in the event a crime was committed from their computer, "a Trojan horse did it". Obviously, careful investigation of digital evidence can help to refute or assert the claims. For example, Internet search history, time-line analysis, and signs of malware infection may all provide relevant information. The Trojan defence has successfully been employed in the UK to clear child porn allegations.

Question: Describe the open WiFi defence, and compare it to the Trojan defence. Hint: if you are not sure, search for information on the Internet.

The traditional approach to the problem of malware is to only run programs that you trust: either using a blacklist (which detects if there are programs that

you really don’t trust), or a whitelist (only runs the programs that you really do trust). Techniques of identifying programs include: signature-based detection, anomaly-based detection, digital signatures, and reputation-based security. There are now sandboxing and isolation techniques that can reduce the damage that applications can cause, although these mechanisms are still typically not used on desktop computers or servers – they are used on some types of phones. You are encouraged to read more about these methods for detecting malware, choosing which programs to run, and limiting the damage they can cause.