IDEAL SCENARIO

- The goal of this talk is to see what can be done if somebody has left their computer logged on and went to go do something else.

Common things people do

- If you ever left your computer on your friends might decide to pull a prank on you and post something ridiculous on your social media or like save and close all your open files or change your desktop background
- If they are not your friend they might try to quickly copy files from your computer, but that's not really ideal because they don't know what files might have sensitive information if any and it would take a long time

What could somebody experienced do?

- This presentation we will show you how to install the most basic backdoor to a computer within minutes and then remotely connect.
- This is a really basic method and your tracks are not covered well from somebody who is experienced with security

What is an operating system backdoor?

- A way of bypassing normal authentication and gaining unauthorized access to a computer

Prerequisites

- If you want to set up a backdoor really quickly, you need to have a few tools prebuilt and available to you somewhere so we have everything ready for when the user leaves the computer
- Portable applications are basically applications that don't rely on dlls, they don't rely on registry settings and they have a very small footprint on the operating system. So basically you just put them on the computer and you can run them.

Starter toolkit for windows 7

- What you have when you have a netcat backdoor is a cmd to the victim's computer
- You want to be able to do everything in that cmd window like modify files and download other files from the internet.
- gVim has a portable binary, so once you have connection to the target computer through netcat, you can use gVim to edit files
- wget allows you to download additional files from the web through command line (it supports HTTP, HTTPS and FTP protocols)
- And of course you would need netcat, which creates a connection between the target and your computer and it'll allow you to transfer data.

Setting up netcat

- In the command shown, it is running netcat with a persistent listener on port 449 and we are telling it to execute the command cmd.exe…. which is to open a shell
- What this allows you to do is as soon as somebody connects to port 449 netcat will connect and open a command prompt right away.
- An interesting thing to note is that there are 2 versions of netcat
- One with the –e option and one without the –e option
- A lot of antivirus softwares flag the –e version as a virus and you can see why it might do that since you are able to use it to open a shell.
- Another thing to note is that this connection that you make through netcat is not encrypted.
- Also, it is not just you that could connect to this port other people in the local LAN are also able to connect to it.

Before you can connect to the netcat instance

- Before we can connect to the netcat instance, we need to make sure that the operating system lets us connect to that netcat instance.
- First, we put netcat in the system32 folder to kind of be sneaky because system32 has a lot of important files
- Also as an average user without security background if you see that the running program is from an important system folder and not just like your downloads folder or desktop, you are less likely to just delete the program

Starting netcat on system boot

- What this command is doing is adding an additional registry setting to start netcat when windows reboots
- So if somehow your netcat crashed and you lost connection, or the victim was able to find the netcat running and killed the process, once the computer reboots, you can gain connection back once again.

Changing/Adding to firewall settings

- The first rule there is allowing UDP on port 449
- When you are setting up netcat, you allow either UDP or TCP because those are the 2 protocols that netcat is able to use
- The next rule is basically allowing netcat to communicate through the firewall

Current setup

- So far we have netcat on the computer, we modified registry settings and changed the firewall to allow communications with netcat
- The problem now is that the netcat instance has not started yet
- Since we are using windows, here is a visual basic script that will start netcat

Connecting to the victim computer

- Ok so now everything on the victim end is setup and you can connect to it from your computer
- Here we are using netcat to connect via the ipaddress and port

Many things you can do!

- Ok say you go through all their information and there is nothing really valuable that you can gain from it and you still want to exploit the person.
- You can start messing with them by doing things like continuously cycling the capslock every 30 seconds
- You can also continuously cycle the caps lock, num lock and scroll lock to make the keyboard light up funny
- Windows 7 has text to speech so you can even start talking to the person like “hey I see your computer is slowing down”

Windows Fork Bomb

- Here is just the .bat script to do a fork bomb. Basically all it is doing is forking processes and eating up resources.
- This will continue until the computer uses up all the resources and just shuts off.

Moving on to Linux

- The linux version is sort of similar to windows with a few differences

Toolkit for backdooring linux

- Same as windows we have our linux toolkit
- With autossh you can start an ssh session, and if the session was closed or anything it would just restart the session
- Netcat is for the same reason as before with windows
- We also have shred, this is useful if you wanted to remove your backdoor once you were done or you wanted to remove scripts that you sent. It overwrites data multiple times so it is really hard to recover the original data.

GNU netcat is not persistent

- In windows we had to start the netcat once, and anybody that wanted to connect to it would be able to do it
- In linux, the netcat listener only listens for one inbound connection. So once you connect to it, it does everything you want until the connection is stopped or is closed for some reason. When that happens it stops the connection and kills the process.
- To get past that you can just use a do-while loop so the netcat is constantly listening for new connections
- The command is very similar to windows the –l is to tell netcat to listen, -v is verbose mode, -p is for the port number and –e is to execute a command when a connection is established.
- You are doing echo –n because if there was some sort of message displayed when netcat connected, echo –n would take that message and display nothing

Place to hide the script

- If you wanted the netcat to run on boot like windows, you can place the startup script in the init.d directory
- This directory has a bunch of start/stop scripts for various services on your system and apparently it is looked over a lot of times.

Setting up the actual backdoor

- Like in windows we put the netcat in the system32 folder, here we put it in the /usr/bin folder which is part of the PATH variable
- We also modify the ip tables to allow the connection
- The last thing we do is we nohup start the listener.
- HUP is the way the terminal warns dependent processes of logout (stands for hangup).
- This is very important because this disconnects the user from the listener process (the process is no longer owned by the user). If we did not do this, when the terminal that the listener was set up on was closed, the process would just be killed
- Hup.out is where all the information is stored for the nohup'd process
- This relates back to why we did echo –n because if we didn't, the output from connecting to netcat would be in hup.out

Same as windows to connect to netcat

- The only thing to note here is that some versions of GNU netcat, once you connect it shows a blank screen with just the cursor. However if you type commands like ls and pwd you get the output of the actual bash. So you are actually connected to the bash shell but you just don't have the bash prompt displayed, it is just a blinking cursor instead.

Accessing target from outside local LAN

- Everything shown so far was for the local LAN but there are ways to access the target computer from the outside. What you would do is set up a persistent SSH tunnel
- This is a picture that shows what is going on
- You are unable to directly connect to the target computer due to the firewall
- What the tunnel does is map a port locally on that machine, to a port locally on the remote machine.
- If the target allows a port OUT to the private server, then you are able to log into your server and connect to the port on the server

How to prevent/detect backdoor activity

- There are many different types of backdoors and ones that are a lot more complicated, but for the one we talked about, the most obvious 1 is to not leave your computer logged in when you go do something else
- Maybe you got backdoored by some other means via the internet. If it is a beginner backdoor like the 1 in this presentation, you can use process explorer to see the nc.exe running.
- TCPview to get more information about the connection.
- Do not give the main user admin privilege (so they cannot modify the firewall settings)

View connections on linux

- Netstat –lptun will show you the connections that are up right now
- As you can see we set the listener to listen on port 445 and it is showing that netcat is listening on port 445

Modifying firewall (windows 7)

- Useful to change firewall settings if backdoor is detected.