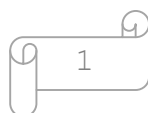


Useful Knowledge
For The
Level One Technician
by
Tim Sayre

Draft version

Windows - Networking w/o all pics

230424



Introduction

I started this book to cover the areas of my training program where I was having trouble finding free material that covered the topics I needed to present. One of the free books was done in such a way that made me think, "I could do that" and here we are. I am going to incorporate some of the material I found in order to save me some time and to prevent me from reinventing a whole course on computer science, but much of the material I found goes in a different direction than I want, so it shouldn't have very much that isn't original.

Since the book was written to accompany a class, the progression will follow the course outline and may not make sense when taken on its own, although it should be a great study guide for CompTIA's A+ certification. While most of the material is original, some is drawn from other sources. Most of the other sources are public domain or creative commons or other free license, but some has been used with written permission from the authors. I will attempt to identify all of those at the time but check the licenses yourself if you plan on using the material in case I missed something. If you use this as a reference, make sure you reference the correct work.

Every attempt has been made to verify all of the information but since the course has not been taught the first time yet, there may be some mistakes. If you find any, please send them to me at tim at kitswv dot com. If you have any suggestions for things you think are relevant that I've omitted drop me a suggestion at the email address above.

I am releasing this book in draft versions as I complete each section. This one is a Windows version with Appendix 1. I am working on the networking section and will release that as Draft 3, Draft 2 being with the placeholder pictures replaced with the actual pictures coming up as soon as I can get the pictures taken. If you have any trouble understanding any of it or have any suggestions for improvement or find errors, use the contact info above to let me know. If you want on an email list to be notified when each draft is released, send me your email address. It will never be shared with anyone without your expressed written consent.

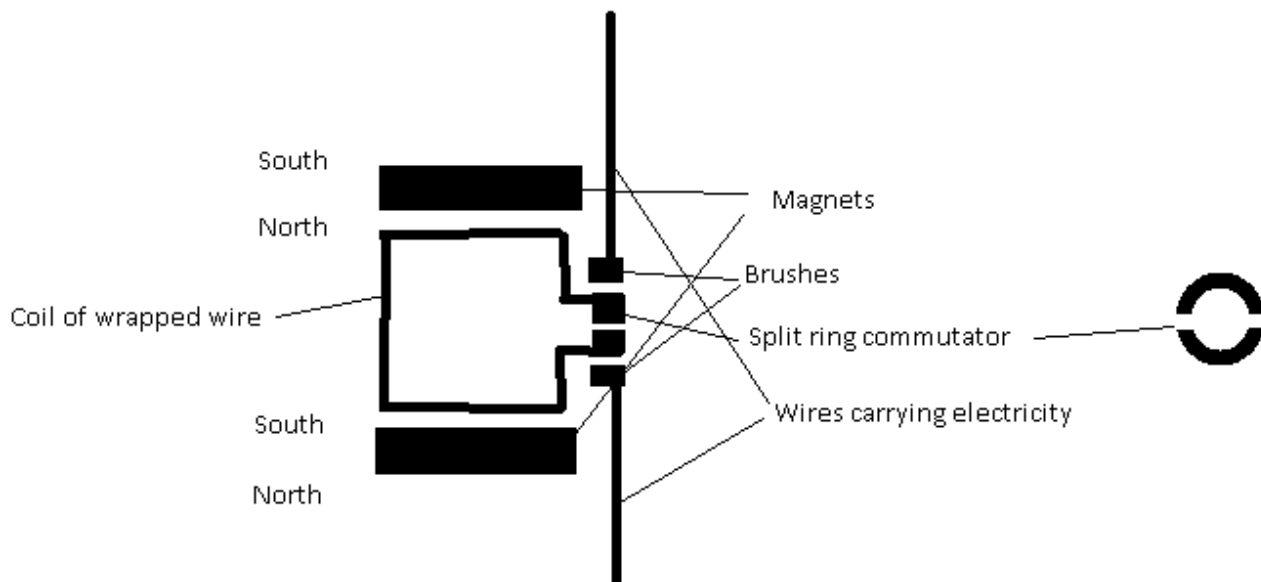
Basic Electronics

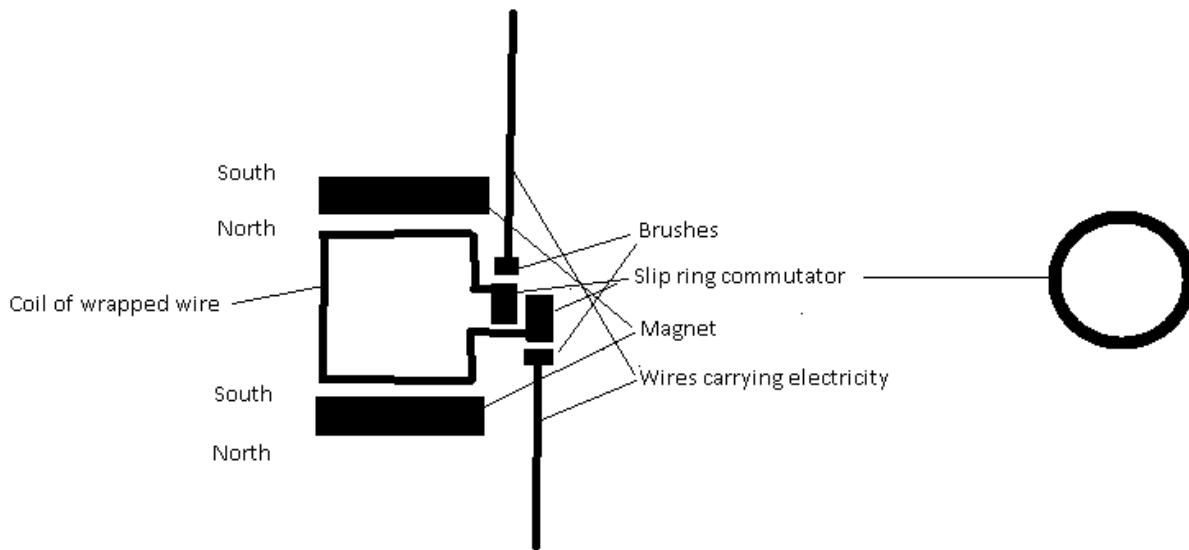
Theory

Most electronic devices work with DC (direct current) electricity while electricity is supplied to them using AC (alternating current). If you apply normal household current of 120V AC directly to the motherboard of a computer, it would fry most of the components. They get around this by using a transformer to convert the AC to DC. In order to understand it better it helps to know a little bit about how electricity is generated in the first place.

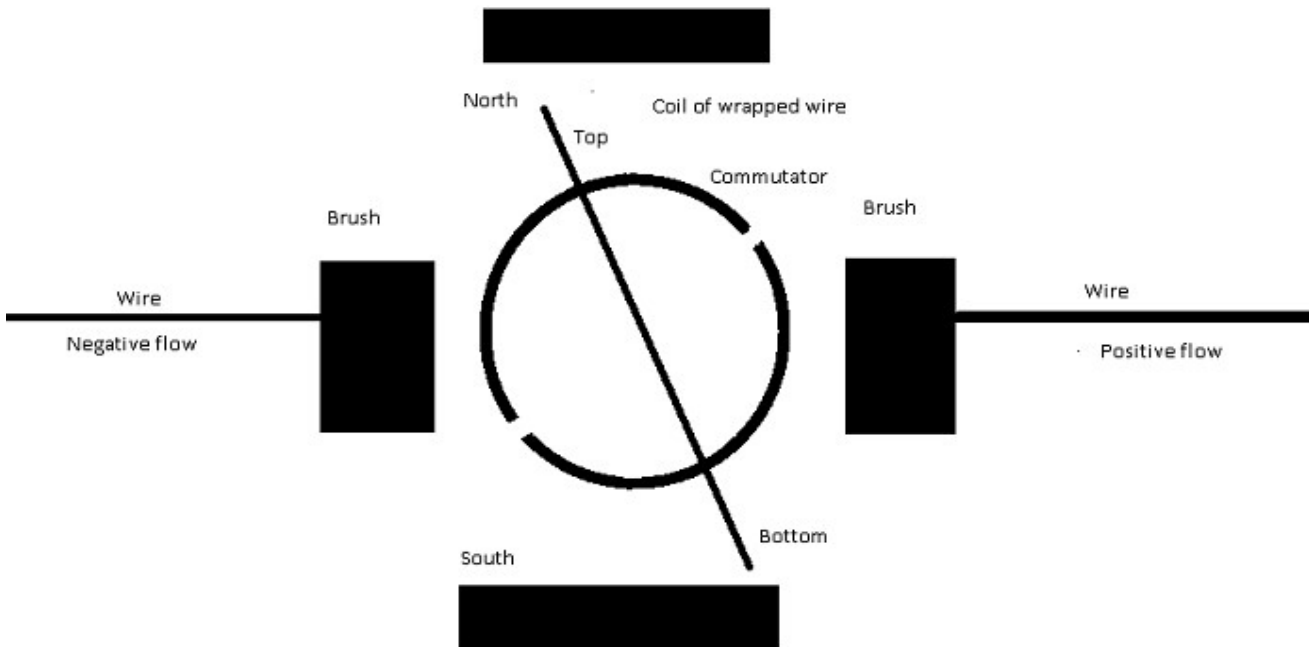
Generating Electricity

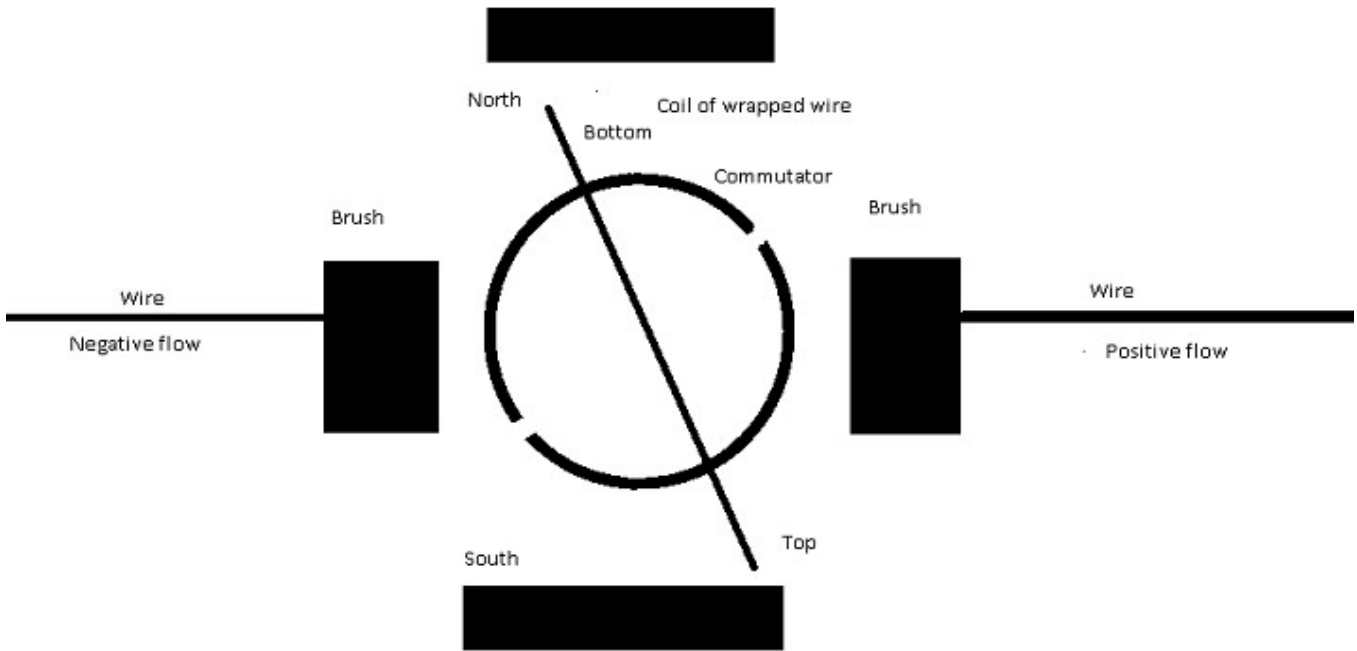
Magnetism and *electricity* are very similar forces. If you take a copper wire and pass it through a magnetic field, it creates electric current by making the electrons move along the wire, if the wire is connected to a load on each end. We use magnetism to create electricity by taking a coil of wires and moving them through a magnetic field made of opposing poles of a magnet. DC current is made using a split ring commutator, while AC current is made using two slip rings.



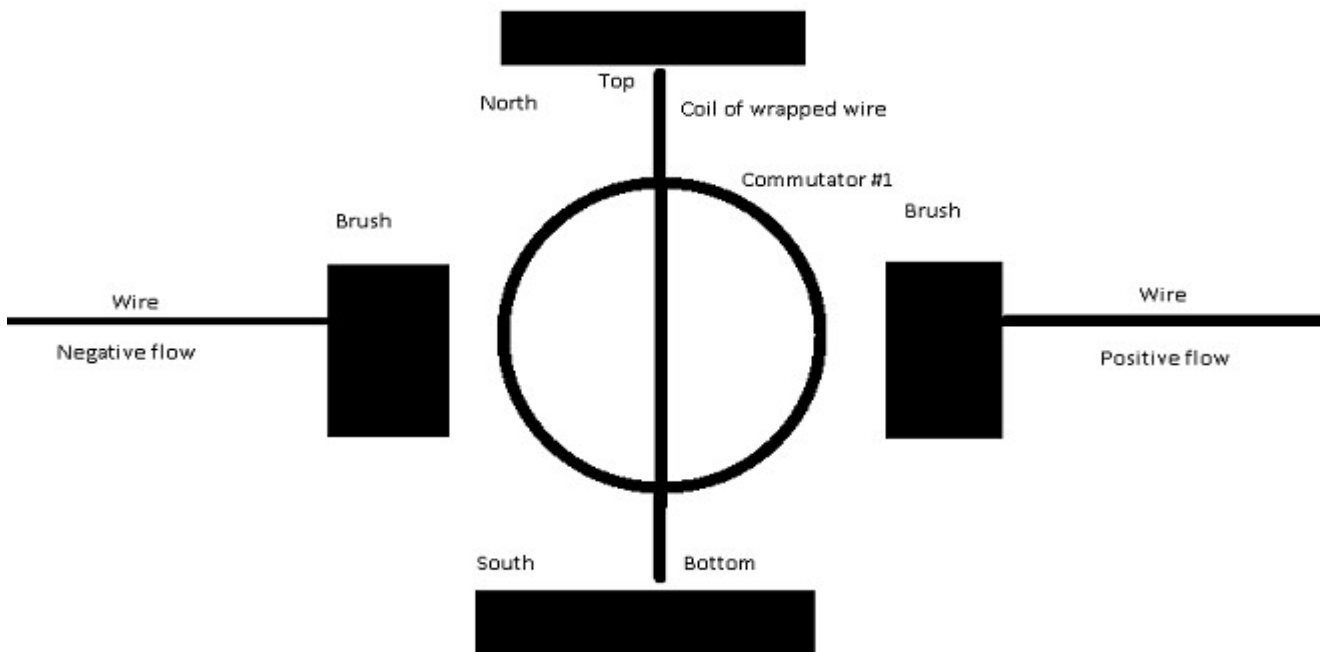


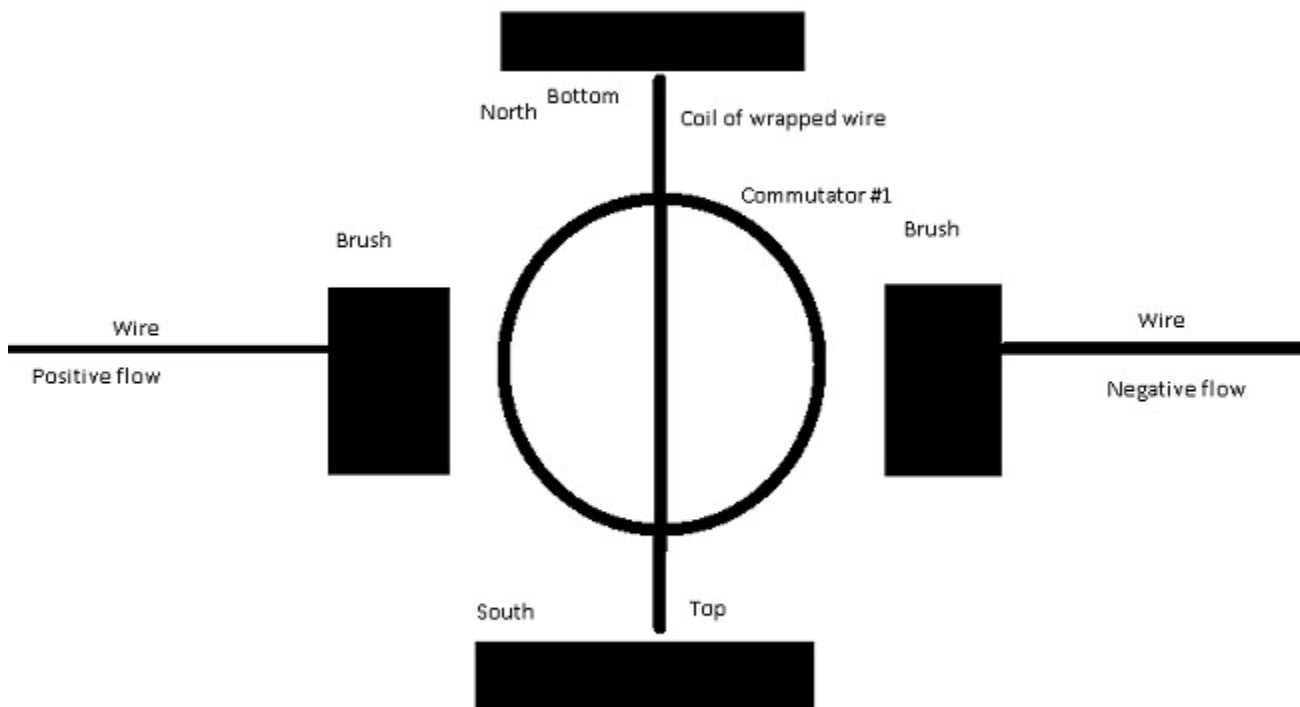
The commutator rotates the rings between two brushes which transfer the electricity generated to the circuit. The split ring allows the wire to separate the positive charge from the negative, depending on which pole of the magnet the wire is passing by. Each side of the commutator passes one charge only, positive or negative, not both. Notice how the current flow stays the same on each wire regardless of the coil position.





This is why DC only flows in one direction, negative to positive, while AC changes directions. AC has two rings and attaches one to each end of the load and when it passes through the opposite poles of the magnetic field, the current reverses. Notice that the current flow changes, or alternates, depending on which side of the coil is under which pole.





The motor turns 60 revolutions per second, which is measured in *Hertz* (Hz). This is why AC current reverses *polarity*, the direction current is flowing, 60 times each second which gives it a *frequency* of 60 Hz.

Now forget all that. Just kidding, mostly. It was all background and you need to know it, but just the mechanics of it, not the details. In fact, this whole section on basic electronics is not much use to a help desk worker in a call center, miles away from a computer. But the basic knowledge needs to be there as a foundation to support the knowledge you *will be* using on a daily basis.

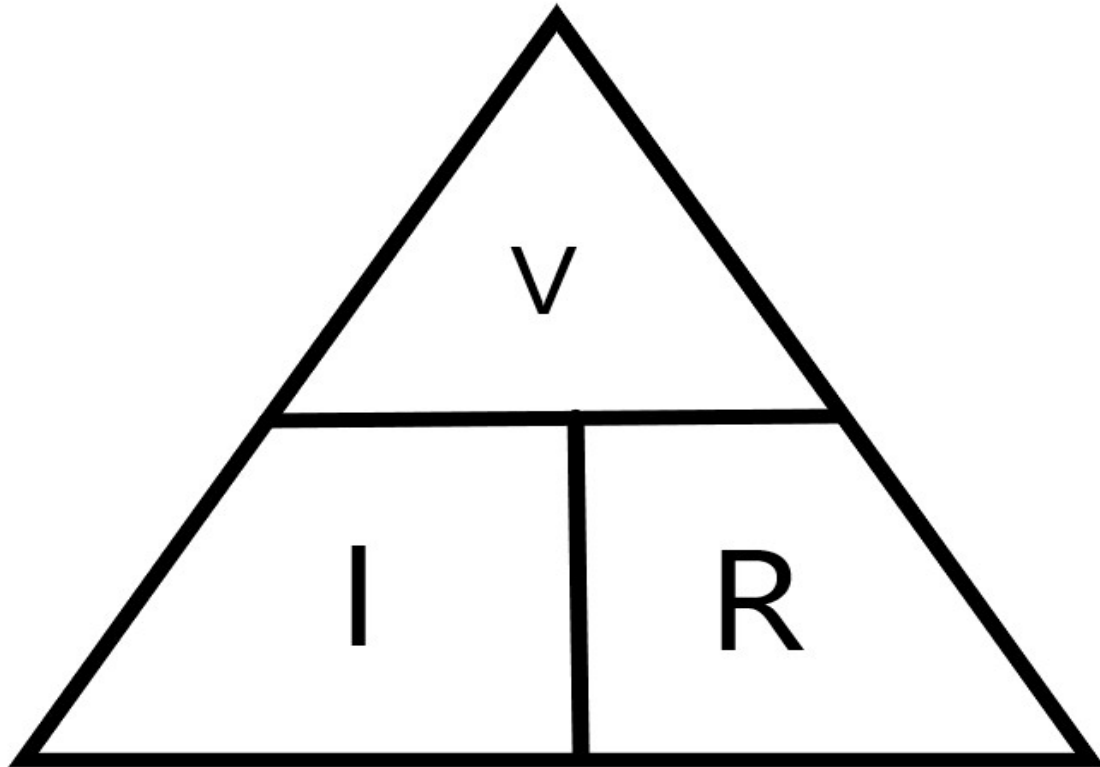
Measuring Electricity

We use different things to measure electricity; *Volts* (V), *Amperes*, aka *Amps* (I), *Watts* (W), and a related unit *Ohms* (R).^{*} Technically ohms measure resistance, but since this is familiarization I am putting it here since it is resistance to the flow of electricity.

^{*} Also, amps are sometimes referred to as A, and Ohms as Ω .

Volts measure the potential difference between two points while amps measure the flow of current, or *electromotive force*. The electromotive force is what makes the electrons move and the current "flow".

Resistance is measured in ohms and slows down the electrons. Watts are a measure of power that we get when we multiply volts by amps. This is useful for knowing how much power your system is drawing to ensure you have a large enough power supply. It can also help you find out if you are drawing too much power for your breakers to handle in your home.



Ohm's law says voltage equals amperage times resistance ($V=I \times R$), amperage equals voltage divided by resistance ($I=V \div R$), and resistance equals voltage divided by amperage ($R=V \div I$).

* You may see * instead of x for multiplication and / instead of ÷ for division.

How it Moves

Without going into too much detail, be aware that everything is made up of atoms. Atoms have a neutron with a neutral charge, a proton with a positive charge, and an electron with a negative charge. The neutron and proton make up the nucleus while the electrons revolve around the nucleus. Positive and negative charges naturally repel each other. electrons are grouped in *valence shells*, or rings, which determine how far they are from the nucleus. The electrons in the shells closest to the nucleus have a higher charge than the ones further out. If there are an odd number of electrons compared to the protons we call those *free electrons*.

When we say electricity flows from negative to positive, we mean that it gives up free electrons along the conductor to complete the circuit. Some materials give up electrons easily and are called *conductors*, while others don't and are called *insulators*. We all know that metal conducts electricity and rubber doesn't, that's why electrical cords are made with metal wire covered with rubber insulation. In a nutshell, the current excites the atoms and electrons are forced along the conductor while the atoms in insulators refuse to give them up.

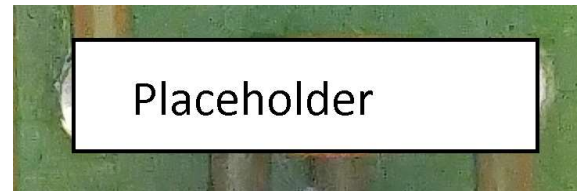
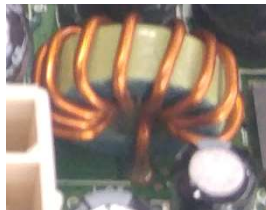
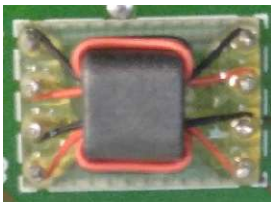
Electricity, along with water and most things in nature, takes the path of least resistance. This is how you get a *short circuit* or unintentional path to ground. That is when something affects the path the electricity is supposed to take and causes it to bypass the actual path and usually involves damage to components. There is a saying that applies here, "Nature hates a vacuum" which means that everything is trying to come to *equilibrium*, or balance. You can see that in your daily life using a glass of water. Fill the glass, then place a bowl over the top turned upside down with the bottom of the inside of the bowl over the top of the glass. Turn it upside down and take the glass away. The water tries to fill the bowl equally instead of keeping its shape. There is a lot more than that going on and I encourage you to explore further.

One of your main concerns when working with electronics is heat. Working with low voltage DC allows you to use smaller wires than used in household currents. When electricity flows it generates heat. If the current is too high for the diameter of the wire to *dissipate*, or get rid of, the heat it can damage equipment or start a fire. You can use the formula above to find the values and look up the maximum allowable for the wire you are using. Never leave cords coiled while using them. The heat could melt the insulation and start a fire.

Electronic Components

Transformers

A transformer raises or lowers voltage; known as step up or step down. You'll mainly encounter them in power supplies or on power boards. They work through *magnetic induction*. The current travelling through the wires wrapped around the primary coil creates a magnetic field and induces electricity into the wires on the secondary coil where it is transmitted down the wires to the load. If there are more wraps on the primary than the secondary, it is a *step down* transformer, while more wraps on the secondary make it a *step up* transformer. The step up transformer increases power while the step down decreases power.

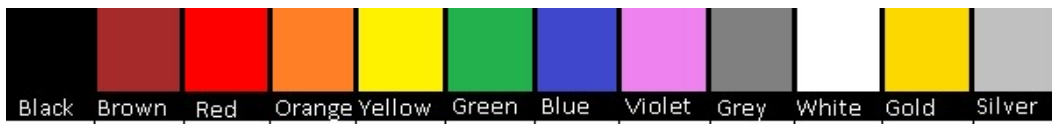


Resistors

Resistors are devices that have a set amount of resistance. They are made from wire, carbon composition, ceramic film, metal film, metal oxide film, and foil. They operate on the concept that some material causes voltage loss through resistance while travelling through that material. Voltage loss also occurs when current travels along wires.



They are marked with a series of colored stripes or numbers for the amount of resistance. They have no polarity, which means they can pass current in both directions with no change. The table below shows how to read the colors to determine the resistor value and tolerance. On a four band resistor, the first two digits represent numbers and the third is the multiplier, while on a five band resistor, the first three digits are the numbers and the fourth is the multiplier. The last band is the tolerance, or what the percentage of variance is. In the first example the tolerance would be 235 Ohms, while in the second it would be 5200 Ohms.



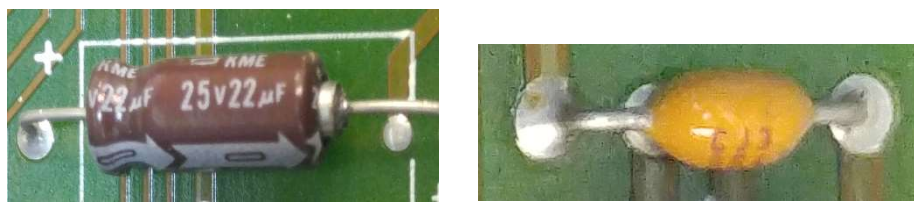
| | | | | | | | | | | | |
|---|-----|-----|-------|--------|---------|-----------|------------|--------|---|-----|------|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | | |
| 1 | 10 | 100 | 1,000 | 10,000 | 100,000 | 1,000,000 | 10,000,000 | | | 0.1 | 0.01 |
| | +1% | +2% | | | +0.5% | +0.25% | +0.1% | +0.05% | | +5% | +10% |

Digit
Multiplier
Tolerance



Capacitors

Capacitors take current and store it for a period of time. They can be made of several materials: ceramics, electrolyte, tantalum, silver mica, or film.



They work by having two conductors separated by a filler material that allows for a large potential difference of electrical charge. Ceramic and film capacitors have no polarity while the others do. They are typically used to filter out noise from DC circuits, audio signal filtering, and smoothing out DC voltages. The capacitor on the left above is an electrolyte capacitor and when they go bad they swell up and sometimes leak fluid.

Diodes

Diodes are made to allow current to travel in one direction, so they have polarity.



LEDs are light emitting diodes and are used for light sources such as flashlights. They are also used in reverse current protection circuits and rectifier circuits that change AC to DC.

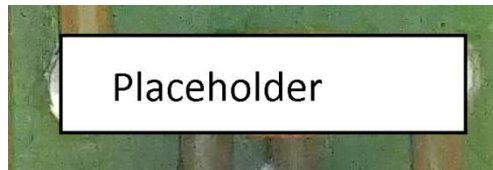
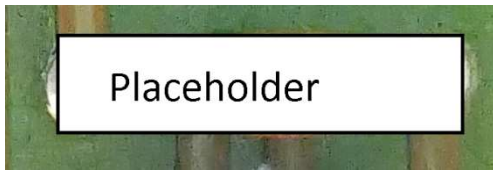


Zener diodes are designed to let current flow backwards when the voltage gets high enough. They are used to protect circuits from static discharge, change sine waves to square waves, and as a voltage stabilizer.

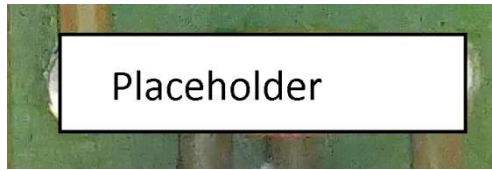
Shottky diodes convert AC to pulsed DC in one direction. They are used to modulate signals, limit signal amplitude, and create logic circuits.

Transistors

Transistors are active components of a circuit, meaning they increase the voltage in a circuit by modulating the power, while the previous components were passive. They are either npn or pnp, negative-positive-negative or positive-negative-positive, depending on how the current needs to flow. They consist of a *base*, an *emitter*, and a *collector* with a *depletion region* separating them.



A small current applied to the base controls the larger current flow from the collector to the emitter on an npn transistor, and from the emitter to the collector on a pnp transistor.



Field effect transistors (FET) use their construction to change the voltage by changing the shape of the channel that allows the flow of current. *JFETs* (junction field effect transistors) use junctions of pn and np to control the flow while *MOSFETs* (metal oxide semiconductor field effect transistors) use the *semiconductor barrier* to achieve a similar result. Transistors replaced *vacuum tubes* in electronics in the 1950's and in the 1960's the use of silicon wafers to make lots of transistors allowed the invention of the *integrated circuit chip*, (IC chip).

I C Chips

Integrated circuit chips come in many different designs and sizes. They are made out of an epoxy plastic shell that protects the silicon wafer containing the transistors that make up the circuit.



The connections are attached to small metal legs that to the boards by soldering directly onto the board or being placed into a socket on the board.

Miscellaneous Components

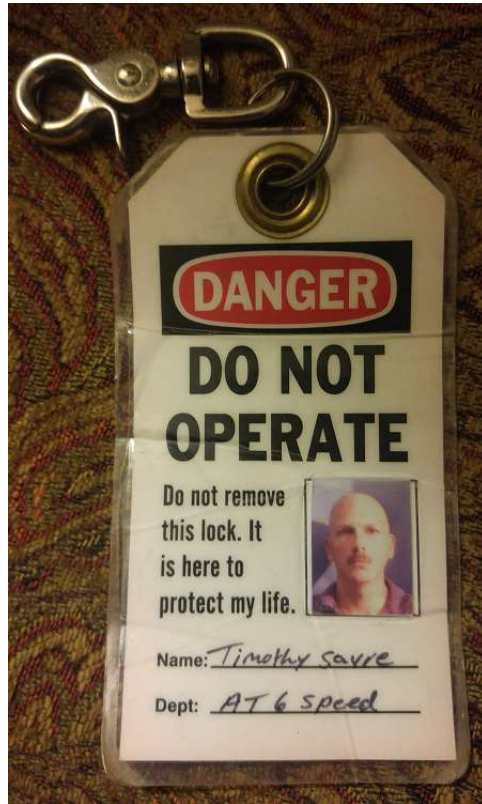
There are other electronic components that I have left out, such as relays, sensors, and inductors, but as stated earlier, this is a familiarization more than anything. You most likely will not be doing a lot of repairing of circuit boards, since things these days are usually cheaper to replace, especially when the repair consists of working with extremely small and delicate components. I encourage you to learn how to solder so that you can continue your training and add another skill to your toolbox, making you more marketable and a better tech.

Safety

"Safety third." That used to be my mantra, but it came from my time in the Army when the motto was, "People first, mission always", so I naturally added my part then dropped the rest when I got out. Of course safety is the most important consideration when dealing with electronics. Not only does electricity hurt when you get shocked and it can kill you, but electronics are expensive and very small amounts of electricity can do a great deal of damage. I wrote a blog post on this very subject [here](#), but this will go into a little more detail on the why's an how's.

First and foremost, make sure to disconnect the power from whatever you're working on. Turning it off isn't enough; you need to remove the power source entirely. In commercial environments, you may encounter large machinery using large amounts of electricity feeding from several

different locations. Lock out tag out, *LOTO*, ensures that you are protected by putting a physical lock or tag on every source of power or stored energy before starting work. When someone sees this tag or lock, they know not to turn the machine on while you are still working on it.



Don't think that just because the power has been removed there is no power to the device. Capacitors are designed to build up a charge and let it out slowly, but when you arc across both leads it discharges all at once and it doesn't feel good. When you unplug it, hold the power button down for 15 or 20 seconds to drain the residual power from the capacitors.

Static electricity can damage some electronic equipment. To prevent it, touch the metal case of what you are working with, or a common ground on the board or case. You want to remove carpet from your work area since walking across it can build up static electricity. Rubber mats are a good idea as they also prevent back strain from standing on hard surfaces. Wearing grounding straps is another good way to prevent damage.

Use insulated tools while working on electronic equipment. TV power boards can carry voltage in excess of 10,000 V in some areas, and I can tell you from experience that you are going to forget to discharge it once and usually only once. In some situations, for some people, that is lethal. Don't take the chance when it's so easily prevented; first, by following the rules above, then by following this as a backup.

They are not very common today, but if you ever work on a CRT (cathode ray tube) monitor or TV, then be aware there are very high voltages in there, up to 30,000V in some cases. This is usually concentrated at the emitter, or base of the tube which emits protons onto the back of the tube and creating the images. It was easy to "burn in" an image on the screen, which is why they created screen savers.

When working on electronic equipment, we use various chemical cleaners and lubricants. Some of them give off toxic fumes, and some are toxic when touched. Always read all the warning or caution labels on chemicals you use, and be sure to follow all safe handling instructions. Keep a spill kit on hand if needed for hazardous chemicals. Always keep your work area well ventilated and clean. Keeping the area clean also prevents trips and falls, so make sure to properly store all equipment, power cords, or other tripping hazards when not in use.

Make sure that you have the proper protective equipment (PPE) on at all times. We mentioned all sorts of dangers that we work around and all of them have mitigations that allow us to work safely. Goggles or safety glasses, gloves, long sleeves, aprons, and all sorts of other things we use to prevent injury but they are only effective if used properly.

On the same subject, be aware of stray clothing or hair that could get caught in moving machinery, or cause an electric shock, or even get caught on equipment. Beware of rings, necklaces, watches or other jewelry which could cause electric shock and hurt or kill you or damage the equipment.

You may find yourself working with lasers, especially in fiber optics. Most laser light is invisible and can damage your eyes before you know it was there. Use laser glasses to prevent damage, and never look directly into a fiber optic cable, even with laser glasses on.

Finally, always have a fire extinguisher and first aid kit on hand. It also has to be the correct type. A fire extinguisher with the wrong rating could do more harm than good, and not having the proper first aid supplies is next to useless. Depending on the types of chemicals you work with and how often, you may need to have an eye wash station or even a safety shower, although that may be overkill for most workshops.

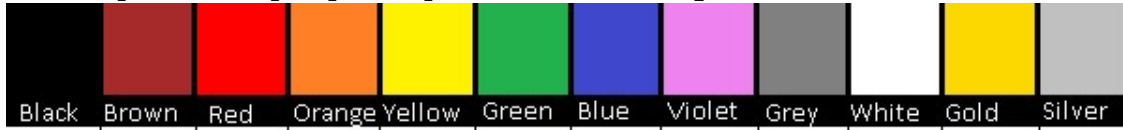
I think your most important safety feature is your common sense. It isn't said enough I think, but nobody has a more vested interest in your well being than you. Be careful out there and stay safe!

Computers

Hardware

Basic Composition

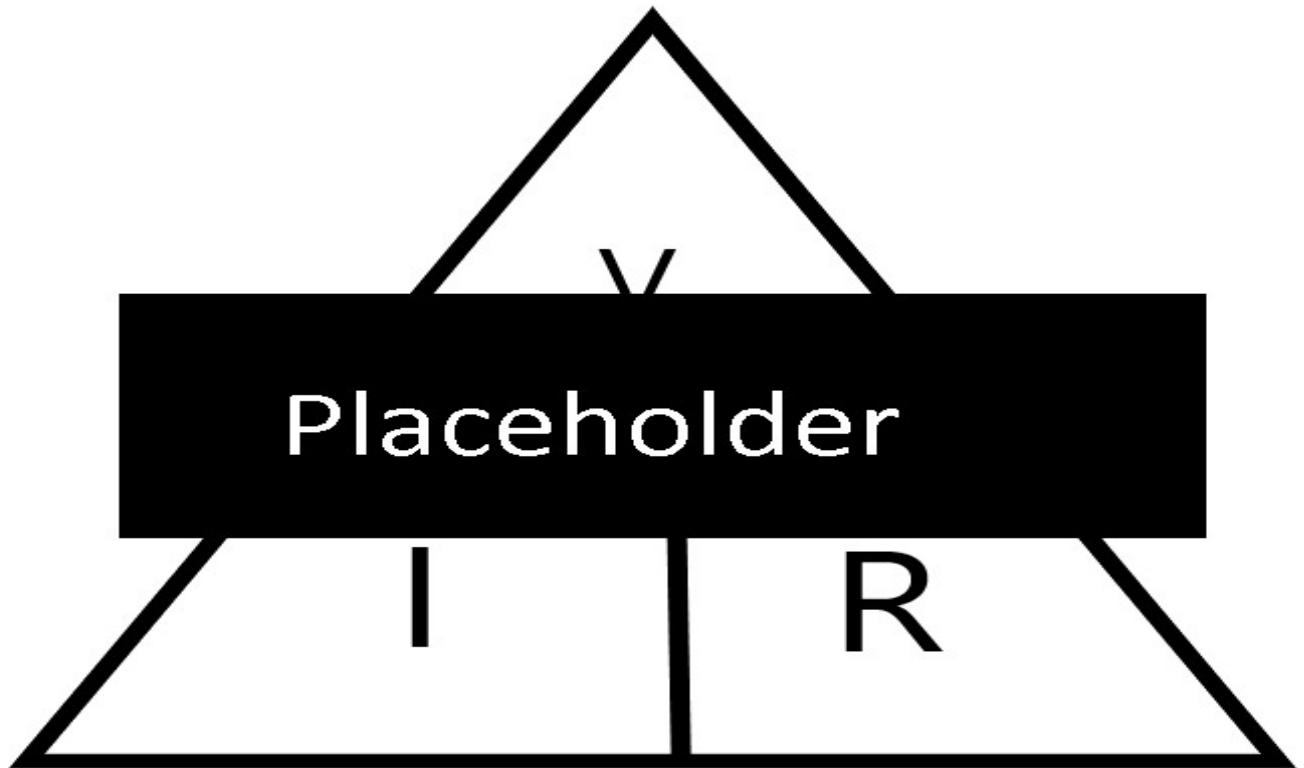
There are several definitions of computer, but we will be talking about desktop and laptop computers commonly found in the home and office.



| | | | |
|---|-----|--------------------|------------|
| 0 | 1 | Placeholder | Digit |
| 1 | 10 | | Multiplyer |
| | +1% | | olerance |



They all consist of a motherboard, central processing unit (CPU), memory modules, hard drive, network interface card (NIC), and power supply.



They are accessed using a mouse and keyboard and have a monitor for displaying what is being done. Some have CD or DVD players or writers,



and some still have floppy disk drives, although these are uncommon today. Technically, the central processing unit (CPU) is the computer, but the rest of the basic components are required for us to use it.

Motherboard

The motherboard has expansion slots for things like video cards and additional memory modules as well as a slot for the CPU.

| | | | | | | | | | | | |
|-------|-------|-----|--------|--------|-------|------|--------|------|-------|------|--------|
| Black | Brown | Red | Orange | Yellow | Green | Blue | Violet | Grey | White | Gold | Silver |
|-------|-------|-----|--------|--------|-------|------|--------|------|-------|------|--------|

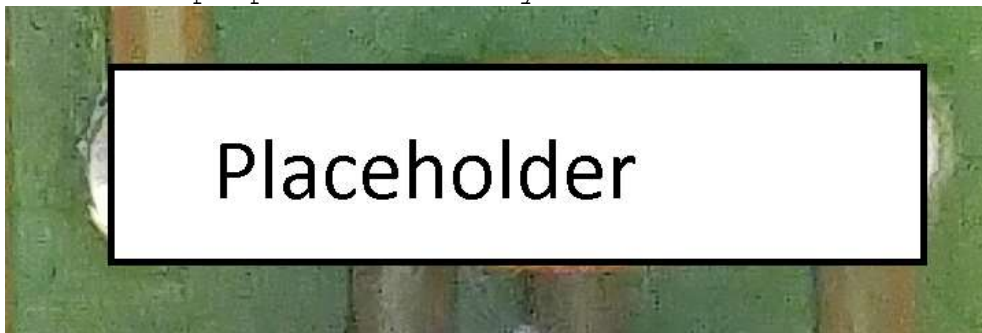
| | | | |
|---|-----|-------------|------------|
| 0 | 1 | Placeholder | Digit |
| 1 | 10 | | Multiplyer |
| | +1% | | olerance |

| | | | |
|---|--------------------------------|---|-----------------------------------|
|  | $47 \times 100 = 4700 \pm 5\%$ |  | $52 \times 1000 = 52000 \pm 10\%$ |
|---|--------------------------------|---|-----------------------------------|

Sometimes the NIC is built into the motherboard, and some have a connector onto which an antennae can be attached to pick up wi-fi.



The motherboard also has additional inputs such as USB (universal serial bus) or SD (secure digital) card slots. Older motherboards have PS2 connectors (personal system 2), and they were color coded green for the mouse and purple for the keyboard.



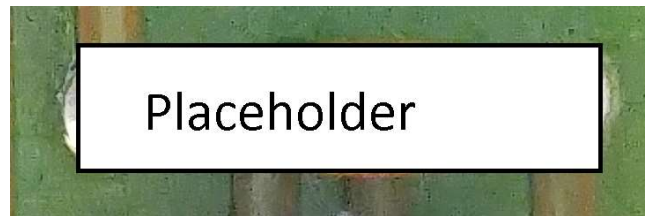
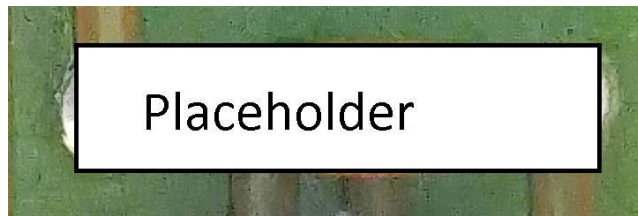
The motherboard has what are known as expansion slots to add accessories, such as video cards, network cards, or sound cards to the motherboard. There have been many different types of connectors for these expansion slots, but PCIe (peripheral computer interconnect express) is almost exclusively used today.

Troubleshooting

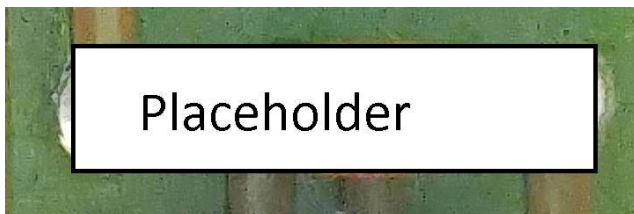
There isn't a lot of troubleshooting you can do on a motherboard, mainly just looking for damaged or burned places, reseating cards and cables, making sure jumpers are correct, looking for damaged components, and verifying everything is connected in the proper place. Jumpers are sometimes used to connect pairs of pins on the motherboard. Most of the components are very small so it isn't easy to tell if they are damaged. Use a bright light and a magnifier to inspect it closely.

CPU

The central processing unit is the brain of the computer. It sits inside of a *socket* on the motherboard with a *heatsink* attached to the top of it. This is to cool the CPU since the construction and size does not allow the easy dissipation of the heat created by its operation. Since the surface of the heatsink doesn't touch the CPU evenly, we use *thermal paste* to allow the heat transfer to be uniform and complete. One thing that can happen with a CPU is the thermal paste can dry up and stop working effectively, allowing random errors or shutdowns to happen.



Almost all CPUs now are created using von Neumann architecture except ARM chips, which use a modified Harvard architecture. ARM stands for advanced RISC machine. RISC stands for reduced instruction set computing and ARM chips are what powers smart phones. They are becoming more popular with PCs after Apple introduced the laptop with the M1 chip. Since ARM chips have a reduced instruction set and are set up differently than other CPUs, they are not only faster, they use much less power and run a lot cooler because of it.



This makes them ideal for applications such as smart phones, web cameras, and single board computers, SBCs.



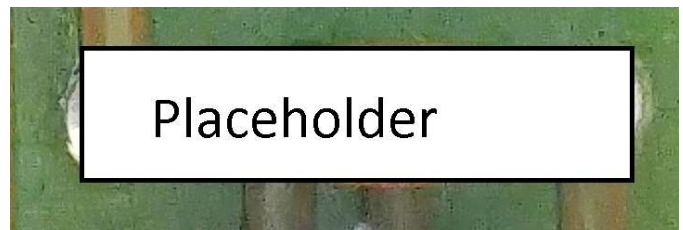
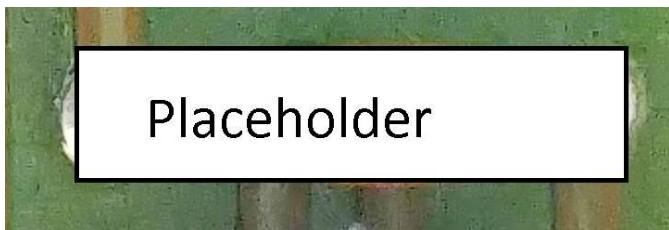
The main difference between Harvard and von Neumann architecture is the way data is handled. Harvard architecture uses separate buses for program data and instruction. Von Neumann architecture uses the same bus for both and this can create a bottleneck for the data transfer. There are various techniques that make up for this with varying degrees of success and each introducing their own unique vulnerabilities and layers of complexity. Intel and AMD desktop CPUs use von Neumann architecture. Since this is just an introduction and has left out a lot of detailed information, as always I encourage you to explore further.

Troubleshooting

There are not many things you can do with a CPU either, as they are built on a nanometer scale in a clean room environment. Before you do anything, look for bent pins on the CPU if there are any. Newer CPUs have the pins on the socket and not the CPU so look there if you don't see any pins on the CPU itself. Next look at the thermal paste to see if it is still there, is covering the entire surface of the CPU, and is not hard, burnt, or brittle. Always use new thermal paste when reseating a CPU after removing the old paste completely with alcohol. If you suspect that a CPU is bad, which rarely happens through normal use, test the CPU in a known good motherboard rather than using a known good CPU in the same motherboard. If the motherboard is what made the CPU go bad, it could do that to the test CPU. You will know the motherboard is bad for sure, but now you need two new CPUs instead of just the one.

Memory

Memory modules are about an inch wide and just over five inches long. They have 168, 184, or 240 pins and a slot in different places to denote different types and to prevent using the wrong kind. They are known as DIMM modules, for dual inline memory module, meaning they use both sides of the pins separately, while older modules were SIMM, single inline memory module, modules meaning both sides of the pin are connected to the same place.



This type of memory is volatile, meaning it doesn't persist between reboots. In other words when you lose power, you lose the contents of memory. All RAM, random access memory, operates this way. ROM, read only memory, on the other hand is non-volatile, meaning it persists between reboots. When adding or replacing RAM, make sure you get the correct type. It could be DDR3 or DDR4, and DDR2 is older but not unheard of. DDR stands for double data rate, and the full name is double data rate synchronous dynamic random access memory. Most folks leave off the SD from RAM when talking about it.



Also, be aware that there are different operating frequencies, 1333 and 1600 MHz are common. It is ok to use different types as long as they all match. You can't use a 1333 MHz stick with a 1600 MHz stick. Desktop computer RAM is also different from server RAM, which has error correction capabilities, and is known as *ECC RAM*.

BIOS

The *BIOS*, basic input output system, aka bootstrap program, is an example of ROM. The BIOS is a small program that starts things up and makes sure

to setup everything to hand over to the operating system. Changing ROM is not easy, and in some cases involves using special equipment and an ultraviolet light. This is where the term "flashing" comes from which means to write new *firmware* to things such as routers, or change the BIOS of a computer. BIOS settings usually have a utility that allows you to update to a newer version much easier.

Troubleshooting

Memory is also pretty small and doesn't have a lot of parts. First, reseal all of the modules by taking them completely out and inspecting them before putting them back. Look for damage to any IC chips or other components and burned places on the board or pins. If there is more than one module, remove them one by one and try booting the computer. It is possible to have more than one bad module, so you can safely use good modules in the bad motherboard to see if it boots as long as you don't leave it on for very long. This can help ensure that you don't ruin a good module with a bad motherboard, but does not guarantee it. Use your own discretion but RAM is usually cheap enough that it doesn't matter as much as it does with more expensive CPUs.

Video

The computer video comes from on board video, which is the inside the CPU itself, through connections on the motherboard, or from a video card plugged into an expansion slot with separate connections. Some CPUs which are common in servers, such as the Intel Xeon series, don't have on board video and require the use of a video card. They are designed for *headless systems*, or operating systems without a GUI, graphical user interface more commonly known as a desktop. Not having a GUI allows for much faster operation and helps reduce the attack surface by keeping out vulnerabilities associated with the GUI.

There are several types of video connectors: VGA - video graphics array, DP - display port, HDMI - high definition multimedia interface, DVI - digital video interface, and a few others that fell out of use.

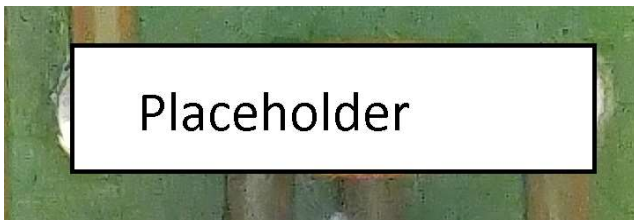
Troubleshooting

Video cards are a little bigger and can be inspected like a motherboard in most cases. With a video issue, you see patterns of colors on the monitor, vertical or horizontal lines going across the screen, dim or fuzzy picture, or white spots across the entire screen, almost like stars. In TVs, which are more like computers now, you can replace capacitors on the boards to fix video issues. The last symptom mentioned, the white

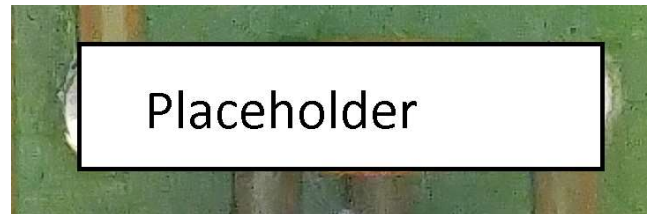
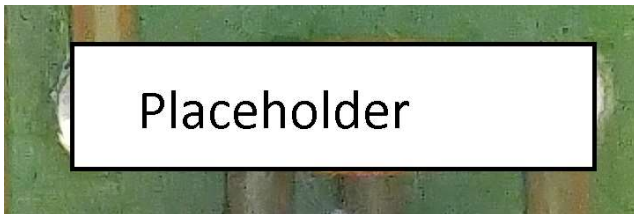
dots, are also a TV symptom. It was a certain type of TV I ran into with a DLP, digital light processing, chip. When the chip goes bad it causes this symptom. For computers, start with reseating the card but pull it completely out and inspect it well first. Spin the fans to make sure they move freely if it has any and apply power to it and make sure they work. You can also test for the proper voltage coming from the power supply but be careful when working with a computer turned on. Most of the capacitors on video cards are too small to replace, but some are not so check them well.

Power Supply

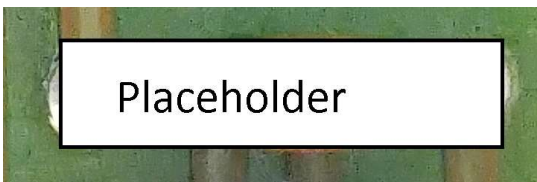
The power supply takes the household current of 120V 60 Hz and converts it to several DC voltages: +12V, -12V, +5V, -5V, and +3.3V. Some have a detachable wiring harness and are known as modular power supplies but most have the cables permanently attached.



The main connector which connects to a 20 or 24 pin connector known as ATX, advanced technology extended. There are also connectors for peripheral devices such as the hard drive, DVD drive, and PCIe slots.



The connectors for the drives are SATA, serial advanced technology attachment, but used to be Molex when IDE, integrated drive electronics, and you may run into them in the field.

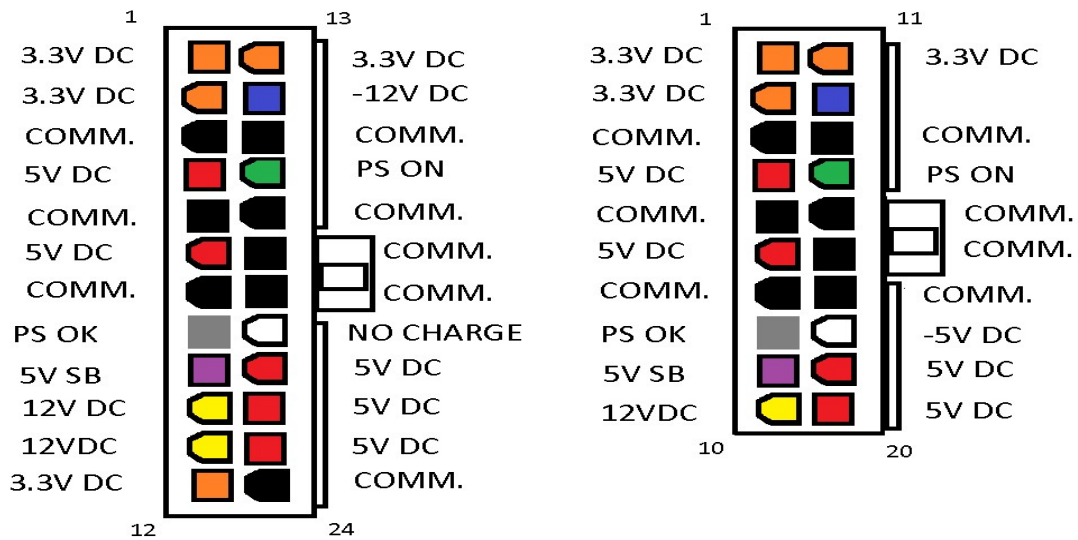


There is usually a six or eight pin connector with yellow and black wires that supplies +12 V to the PCIe slot for accessories such as video cards.

Try to avoid taking a power supply apart. There is usually not much you can do to repair one, and some of the capacitors can be large and hold a charge for a while.

Troubleshooting

To see if a power supply is working, you can use a paperclip to jump start it. Bend a paperclip into a 'U' shape, use a pair of insulated pliers to grasp it at the bend and insert it into pins 16 and 17 on a 24 pin connector, left below, or pins 14 and 15 of a 20 pin connector, right below. When the power supply starts, lay the connector down and use your multimeter to check the voltages to ensure they are correct. You don't need the pliers on this since it won't hurt or even be felt, but don't get in the habit of touching bare wires or conductors. The view below is seen from the bottom of the connector, the part that goes into the motherboard. The colors represent the color of the wire coming out of the top of the connector.



5V SB - 5 volt stand by
 PS OK - power supply ok
 COMM. - common ground
 PS ON - power supply on
 All voltages are DC and positive unless indicated by - in front of the number.

Peripherals

There are other expansion cards for all sorts of things but they all plug into a slot on the motherboard or connect through a port. Most things use USB connectors now, but you may still see parallel cables on printers or PS/2 connectors on mouse and keyboard.

Troubleshooting

If you've been paying attention you probably know this already, but for those who jumped around, skimmed, or don't know yet, reseal the cables and boards, after thorough inspection. Test cables if possible or replace with known good cables. Check for the proper voltages. If possible, use the peripheral in a known good machine. This is the way to avoid a bad motherboard damaging known good devices. For further reference, use the troubleshooting guide located [<here>](#).

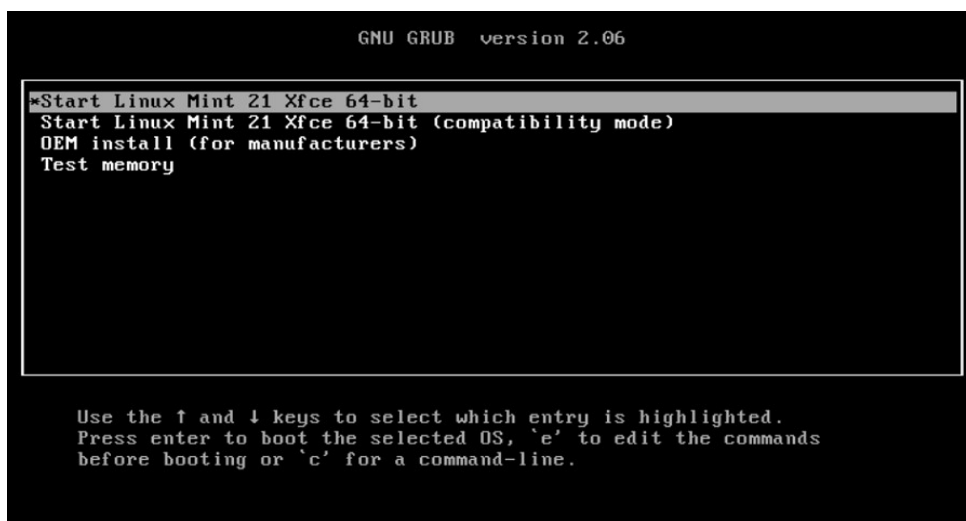
Operating systems

BIOS

We briefly mentioned the BIOS earlier but now it's time to expand on it so you can understand what is happening when you push the power button on the computer tower.

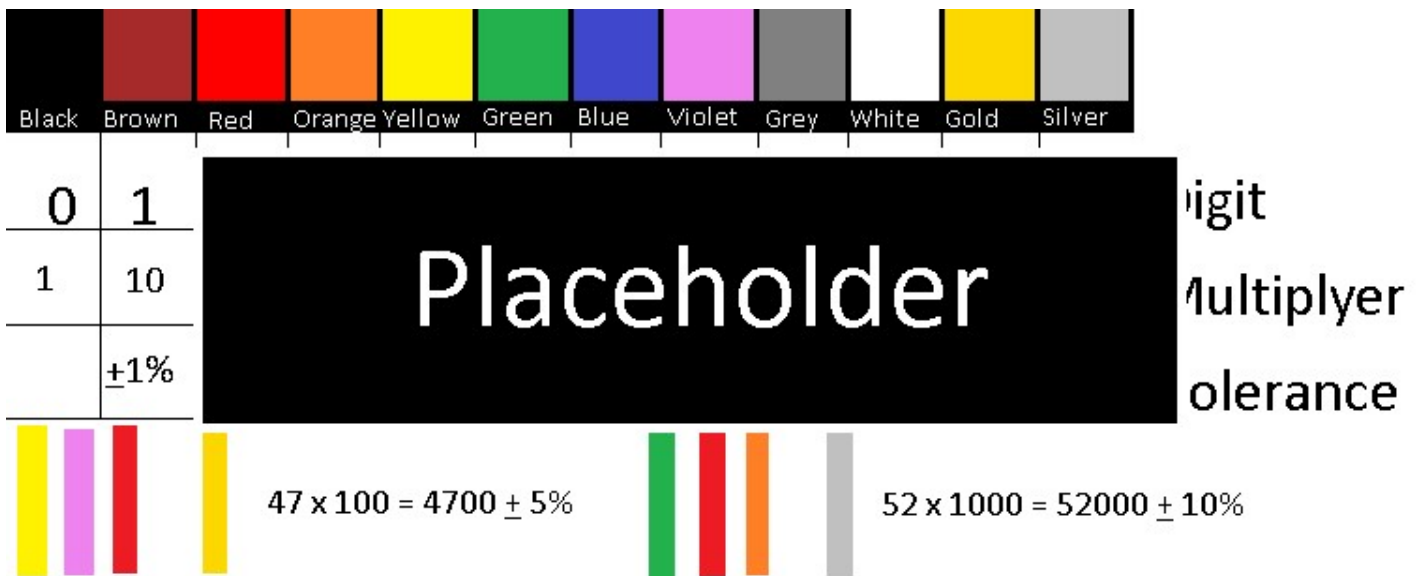
The CPU is what makes the computer work. It consists of a silicon die with billions of logic gates on it that do one thing, or maybe two. They only understand off and on, which we call binary (we'll get more in depth with binary later). Everything more complicated than that is what is known as a *layer of abstraction*, which means that it hides the details of how things work. In other words, we add more complicated things on top of the binary so that we can understand it and work with it more easily. We have assembly language, then the more familiar ones such as C, goLang, rust, python, and many others. We start with English, *human readable*, then go all the way down to ones and zeroes, *machine language*, that executes at almost unimaginable speeds in order for computers to work. While we can still read computer programs, there are several layers of abstraction, so when we get to machine language we find that it is not that easy to figure out the meaning.

But before that happens we need to prepare it for work so to speak. You don't just get up out of bed and start your job and you can't expect a CPU to either. Just like you have to brush your teeth, get dressed, make your bed, eat breakfast, and drive to work the CPU needs preparation as well. The BIOS is a small *bootstrap program*, kind of like a stripped down operating system that allows you to control some of the functions of the system such as networking, sound, certain CPU settings, boot order, and so on. When this starts, it makes sure things are set properly with the configuration file, then hands off to the operating system by reading the *boot sector* and running the *boot manager*.



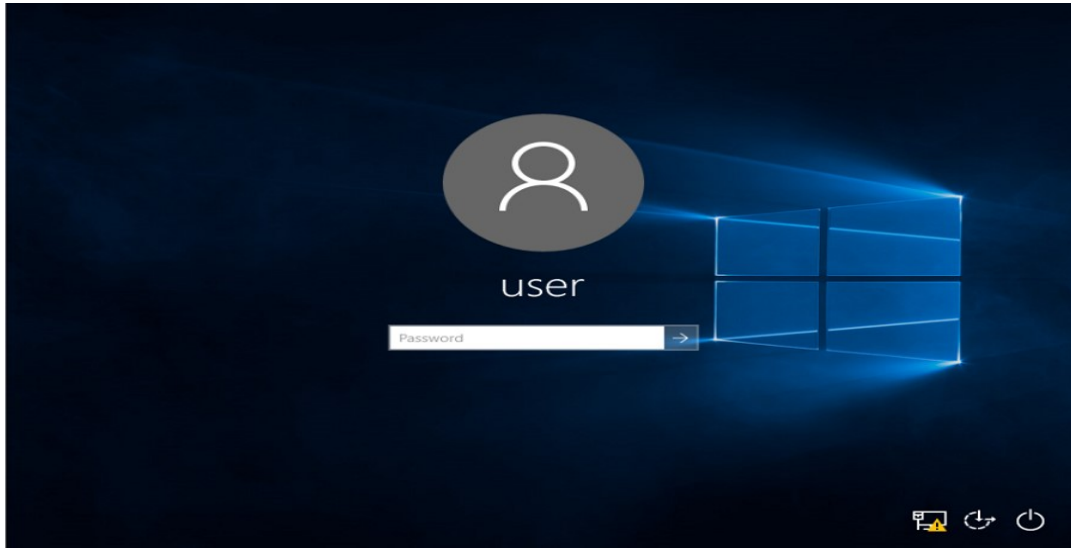
One of the jobs of the BIOS is the POST, or power on self test, and if anything is wrong, gives a beep code. It will beep out a pattern and you look up the pattern to tell you what is wrong. Sometimes this is done by flashing the LED instead of beeping so check the documentation on the device you're working on to make sure.

Before I split up into different operating systems, I want to go over the things they all have in common. They all have a *kernel*, the core of the operating system, and that is how the software uses the hardware. They also use a file structure, although they are all different. Apple and Linux (or GNU/Linux if you want to get technical) are very similar with their file structure where everything is in folders but Windows uses a *hive system* called a *registry*. They also all have a GUI but can all be ran *headless* in server environments and the regular versions of most Linux distributions or distros. They all have two types of users, administrators and standard users. They all produce *logs* which tell you when things happen, including errors. All operating systems can run *scripts*, or small programs written in the terminal or command line using a scripting language, like Bash, which stands for Bourne again shell. They can also run programs, which are more complicated than scripts, although they are not interchangeable between operating systems due to the differences in file structures and other things.



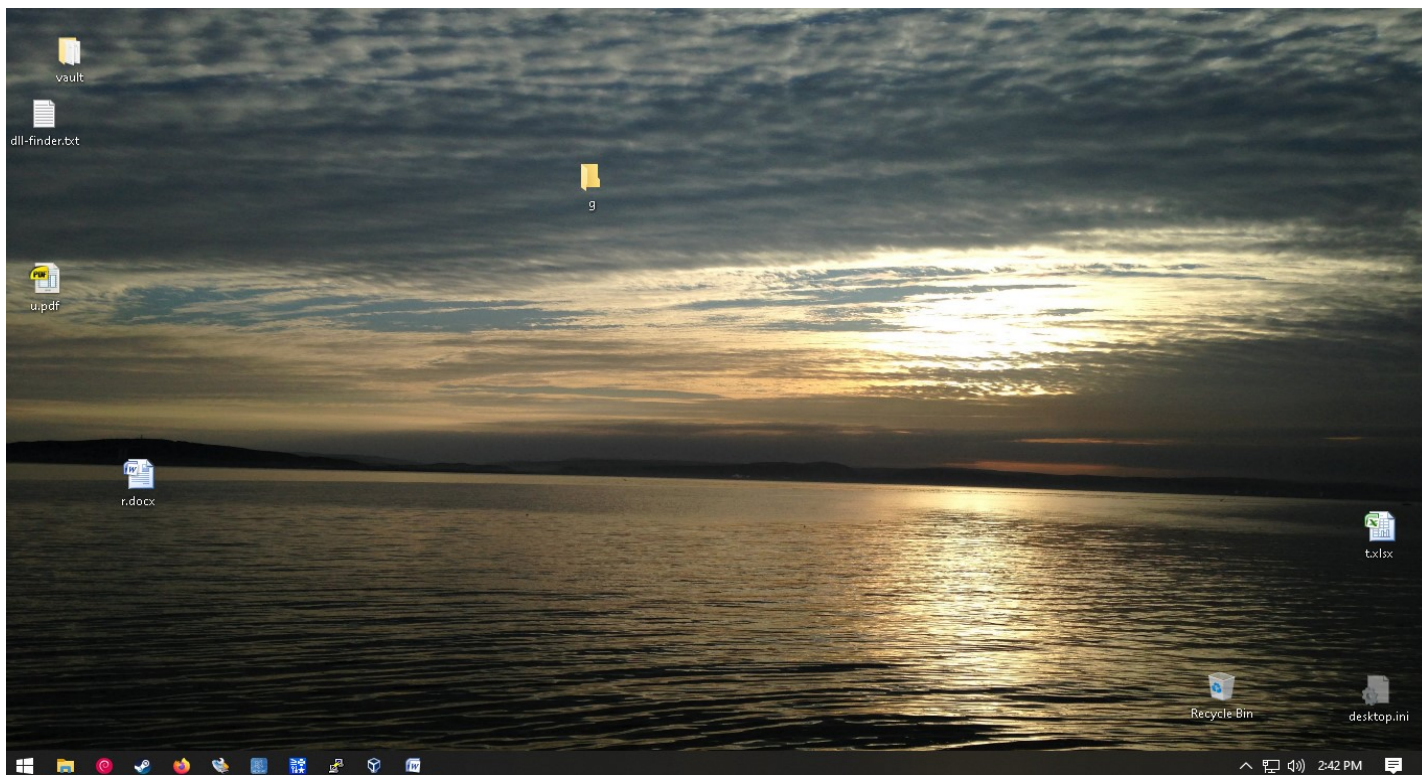
Windows

I'll start with Windows since it's the most common operating system. It is *proprietary*, meaning that the source code is private and not available to look at. That doesn't mean there is a shortage of documentation, it's just that you get what Microsoft wants you to have as opposed to *open source* operating systems like most versions of Linux, where you can not only look at but contribute to the source code if you know how to program.



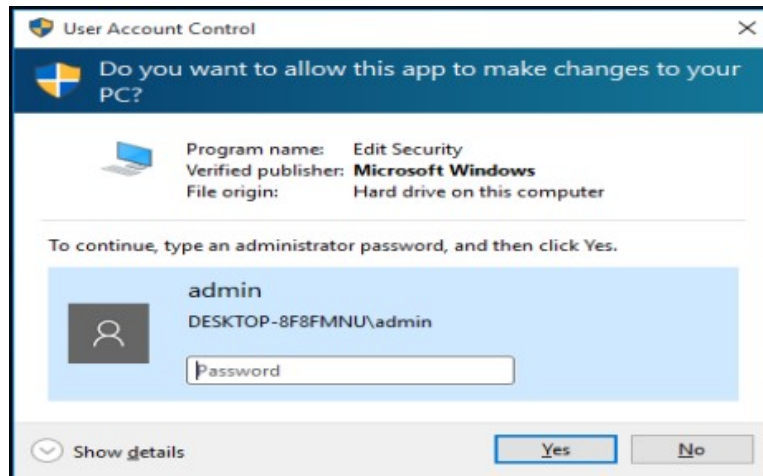
You're most likely to be familiar with Windows and maybe Android but mobile devices are outside the scope of this book. When you start your computer, you may see the splash screen of the BIOS, during which you have a small amount of time to hit a key combination that allows you to enter the BIOS configuration screen as previously discussed. When you are ready to login, you will see a screen similar to the one above. After entering your password, you will enter the GUI.

On the screenshot below, notice the *icons*, or small pictures scattered around the picture and a line of them along the bottom. The row along the bottom is called the *taskbar* and is customizable as well as moveable. You can place it on either side or the top of the desktop or make it auto hide when not in use. On the right side of the taskbar is the *notification area*. The arrow pointing up hides any icons not shown and is a drop-down box (up in this case) that shows them when it is clicked. There are icons for the internet, sound, the time (when you click it a calendar opens), and the message icon. On the desktop you can see folders, files, and the *recycle bin*, where files go when you delete them to give you a chance to decide if you really want them deleted. Notice the desktop.ini file in the bottom right. It is a hidden file, a configuration file that you normally don't see unless you show hidden files. Also, notice I like to use pictures as desktop backgrounds.



The four white squares on the bottom left is the Windows logo and in this case, the start menu. This is where you click to get the list of applications installed, and also what comes up when you hit the Windows key. It has a Windows logo and is between the ctrl and alt, control and alternate, keys on the bottom row of your keyboard.

User account control, UAC, is how Windows handles privileged access to the computer. I would say it is also part of how they approach security and in a round-about way I guess it is, but they dropped the ball by not making the standard user the default like all major Linux distros and Apple does. You should create an administrator or admin account for working on the computer and a standard user account for daily use and normal operation. Standard users can make changes to the appearance and other things related to the user logged in, but not the system itself. When you try to make a system change, the following box pops up asking for your password:

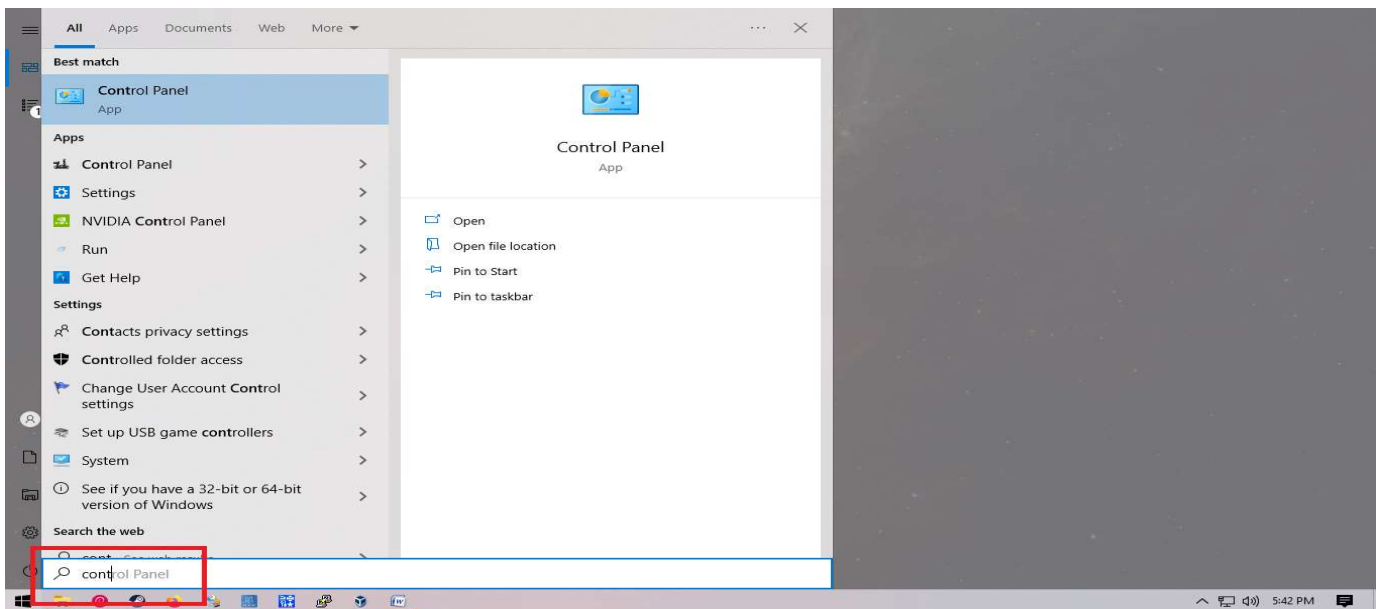


While UAC is active, it is working with a *secure desktop*, which is a Windows feature that only allows system applications to access it. This helps prevent unauthorized applications from running in this secure environment.

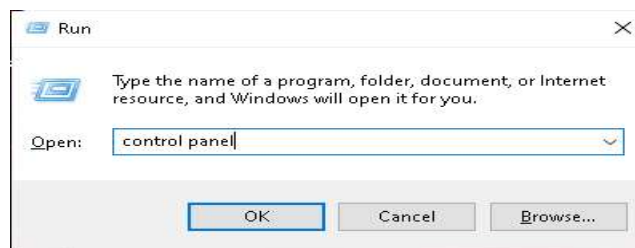
You need to have admin rights for system wide changes including installing software. This is similar to the *principal of least privilege* which says that you work with the account that has the least amount of privilege needed to do the job. When you get infected with *malware*, or malicious software, it has the same rights as the user logged in. If it is a standard user that needs admin permission to make changes to the system, so does the malware. While it is possible to bypass this, not all malware has this ability and it will stop a significant portion of attacks. I have heard up to 90% but cannot find any reliable information to back this up. Whenever you are setting up a new user and have no company policy to go by, for added security, make them a standard user. We will discuss malware further in the software section.

Whenever you do a Windows install do a clean install and not an upgrade. This will save you problems later. As you install, choose custom settings and turn off all of the privacy intrusions and security vulnerabilities Microsoft chooses to turn on by default. Create the admin account first, then the standard user. If you are setting up a lot of computers, write a script to do the configurations, or set up one computer and clone the rest. You can buy programs like Acronis or use open source programs like Clonezilla. We will be going over both methods later.

Most system configuration is done in the control panel or settings. There are multiple ways to get to both of them, but the easiest by far is to hit the Windows key on the keyboard and then start typing. In Windows 8, all you had to do was start typing.



Before you get the whole word typed out, you will see it appear, and then you either hit enter if it is on the top line, or hit the down arrow key until it is highlighted and then hit enter. The next easiest way is to hold the windows key down and hit the letter 'R' which brings up the run box, then type the whole word or phrase in and hit enter.

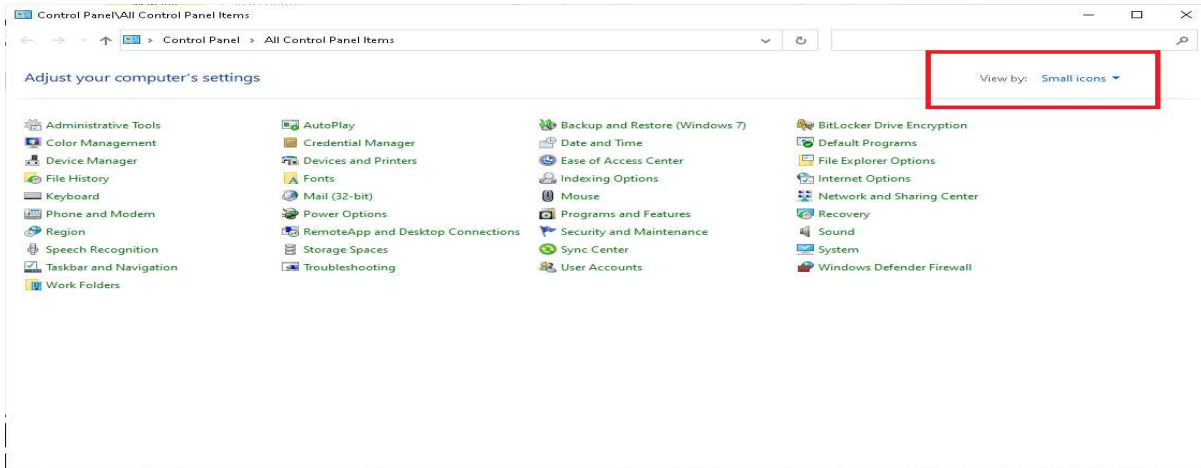


I mention this because sometimes the search box does not appear when you hit the Windows key, but since you have to type the whole word instead of just a few letters, it will be the secondary method of accessing things. The other methods involve using the mouse way too much so I will leave that to you since I prefer keyboard shortcuts. Not only is it much quicker, it gives clients the impression that you are an expert in computer usage. I am including a list of system utility commands at the end of the book.

Control Panel

I'm going to start with the control panel for two reasons. First, I'm more familiar with it since I started working on computers before settings came along. Second, since settings replaced the control panel, it may end up getting dropped from curriculum altogether in some places.

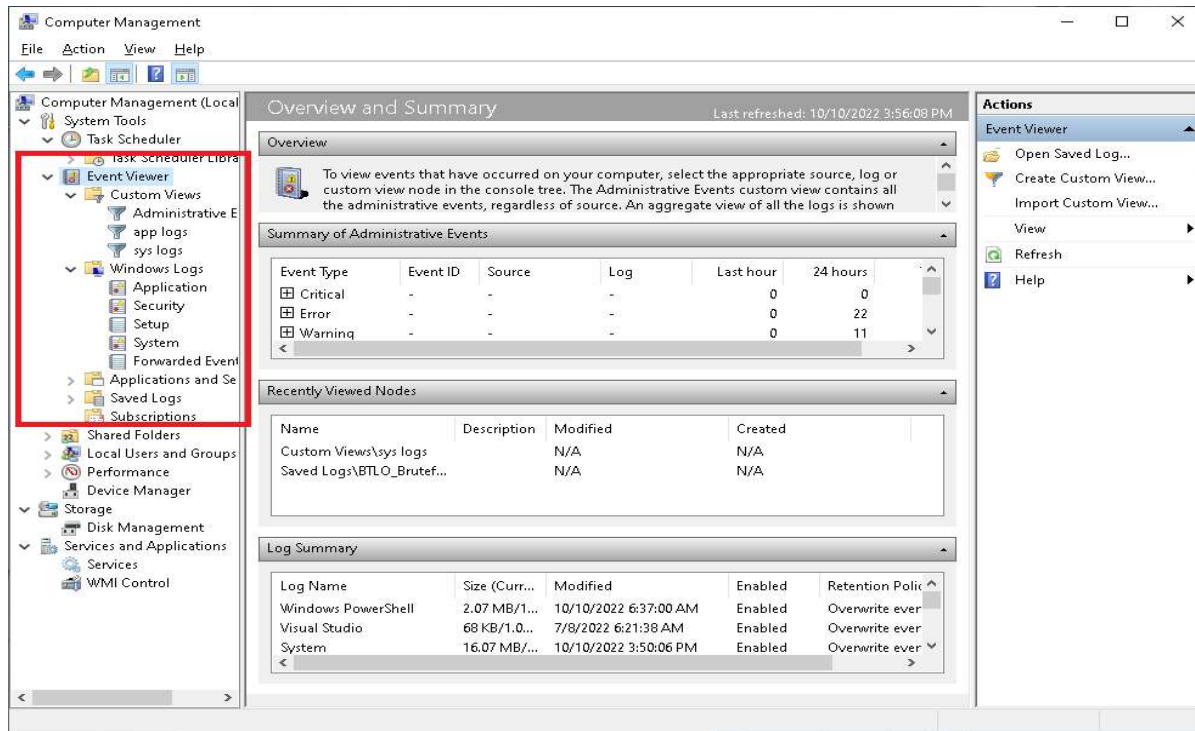
When you open the control panel, it defaults to the category view. I always change it to the small icon view, so that is how I will refer to it. You can choose the view from the top right corner where it says "View by:" just below the search bar.



Administrative Tools

This section has shortcuts to a lot of system utilities. There are a few things in here that I use other methods to get to, but that's mostly by typing. If you need to do something with the system and forget how to get to it, start here. If that doesn't work, hit the Windows key and start typing. In fact, most of the items on this list would be found easier by typing the name of the utility instead of the control panel in the first place, but you have to remember the name, so it's a draw.

I've never used component services and when I looked at my computer nothing was there that I could tell. Computer management, on the other hand, I use all the time, almost every time I troubleshoot a computer. The keyboard shortcut for this is *compmgmt.msc*.

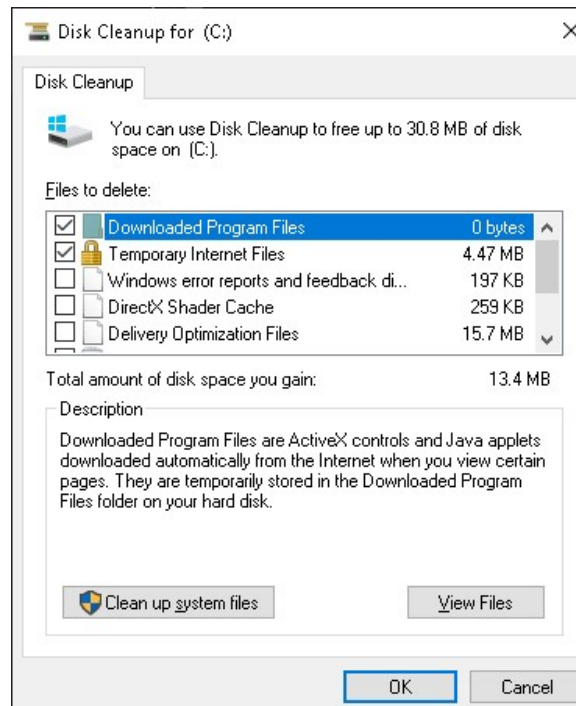


When the event viewer, shortcut `eventvwr`, in the left pane of the window is expanded as in the picture, you can see the folders for custom views and Windows logs. Since a log entry is created every time something is done, you end up with thousands of information logs you need to filter out when you look for errors. You can create a custom view in the application and system logs to show critical, error, and warnings which are the ones that you want to look at first. When you find the errors you are looking for, you can use the timestamp to search the unfiltered log for information entries to help you figure out the cause the error if it isn't immediately obvious from the error message.

When you start troubleshooting, another reason you want to start here is because the computer management console has several of the administrative tools included.

Disk cleanup is one tool you want to remember and it is located here. It calculates how much space you can free up by deleting files. If you click the button that says *Clean up system* files*, it calculates Windows update files and other system files that can be deleted safely.

* When you see a letter underlined, hold the Alt key down and press that letter to open that menu. It is a Windows shortcut that predates the mouse and allows fast operation of a computer using the keyboard only as I mentioned earlier.



We already talked about the event viewer and will go into that more in depth the troubleshooting section.

The local security policy, shortcut *secpol.msc*, is where you would go to look at password complexity and account lockout policy, but on most home computers you don't worry about that. In corporate environments you could use it, but more likely you will see them using active directory and a mixture of Azure and Office for authorization.

The performance monitor, shortcut *perfmon.msc*, can also be accessed through the computer management console. It doesn't have a lot of information for you but there is a link to the resource monitor that does.

Print management is another one I haven't used. This is for use when you have printers to be shared with different levels of access and across different user groups. Any time I have seen printers managed in a corporate environment they were managed through the print server. This looks like some sort of home usage adaptation of that.

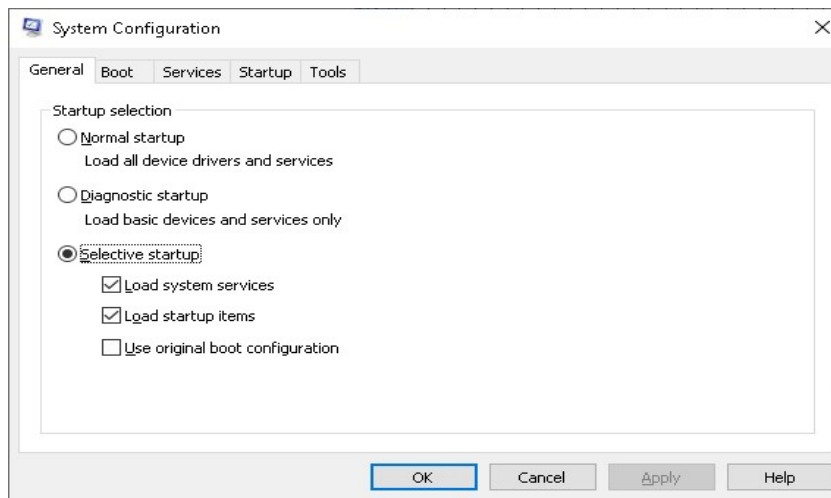
Recovery drive allows you to create a RAID (redundant array of independent discs) type of storage system with extra hard drives or even large flash drives.

Registry editor, shortcut *regedit*, is a place where you will spend some time. You can find and fix all sorts of problems quickly here, but always make a backup copy first since you can brick your system quickly as well. There is no confirmation that you are doing damage and you may not find out until after rebooting.

Resource monitor, shortcut *resmon*, has a lot of good information for you that you can use for troubleshooting.

Services, shortcut *services.msc*, is also accessible in the computer management console. This is where you go when you need to stop, start, or restart a service. You can also get information on the services here.

System configuration, shortcut *msconfig*, is a very important troubleshooting tool. It is a console that controls the system configuration. It allows for diagnostic startups with selections for individual services to be disabled as well as all non-essential services to be disabled at once. This allows you to isolate a problem with a corrupt driver loading at startup.



System information is self explanatory, and one place you can find this information.

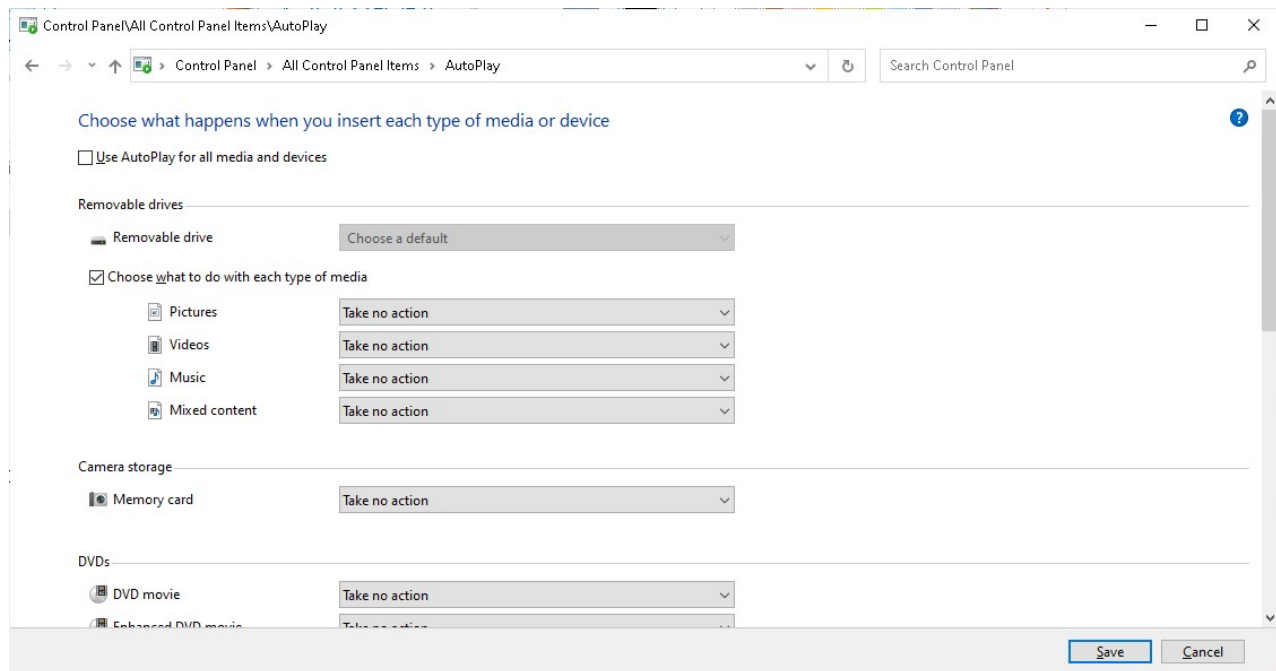
Task scheduler is also self explanatory. This is where you go to set things to be done, like having a certain program started at a certain time, or schedule a backup. It is also a place where malware can gain *persistence*, meaning it can come back after a reboot.

The Windows Defender Firewall with advanced features is one of the ways you control the Windows firewall, which is a software filter that controls access to the internet. This firewall has advanced security, and is where you go to set and edit the individual rules that make the firewall work. When dealing with firewall or internet access issues, this is the place you want to start looking for issues or conflicts. Firewalls will be covered in more detail in the networking section.

Windows memory diagnostic is where you go to schedule a diagnostic to be run on the next reboot. Windows can't test memory while the system is running.

AutoPlay

For security purposes you should always disable autoplay. This gives you the option to use autoplay or configure each option manually. Check the box for 'Choose what to do...', click inside the first box, and hit the letter 'T' (it changes to "Take no Action"), then you hit the Tab key and hit "'T' again, and repeat until you get to the bottom, then hit Tab once more until the 'OK' button is highlighted, then hit enter.



That allows you to make the changes much more quickly than using the mouse. It may not seem like much right now, but if you are working at a call center, one of the metrics they use to evaluate how well you are doing on the job is average talk time. The quicker you can get off of the phone, the more calls you can take which equates to more money being made. We will cover this in more detail in another section.

To expand on the security reasons alluded to earlier, imagine you have a USB flash drive that has malware on it. Of course you don't know it, it has no label that says it's infected,



but if autoplay is set to open files when they are inserted, then you've just infected yourself.

Bitlocker Drive Encryption

It is not something you will use troubleshooting but is a good idea to turn on for the added security if the computer is stolen or accessed without permission. Caution is needed when using encryption however, as your data is irrecoverable in certain circumstances when the password is forgotten. If a backup was not made of the encryption key, there is no way to recover the data⁹. This means that backups are essential. Care is also needed when storing the backup of the encryption key. You shouldn't run into that much because home users don't use it much and enterprise users are usually managed by an IT department.

Credential Manager

The credential manager does what it implies, it stores user credentials. It also stores credentials for various services such as email clients, so if you're troubleshooting a login issue you may want to look here.

Date and Time

If the date or time is wrong you can have errors when trying to connect to secure websites. It will say things about the certificate being invalid. You can change them here.

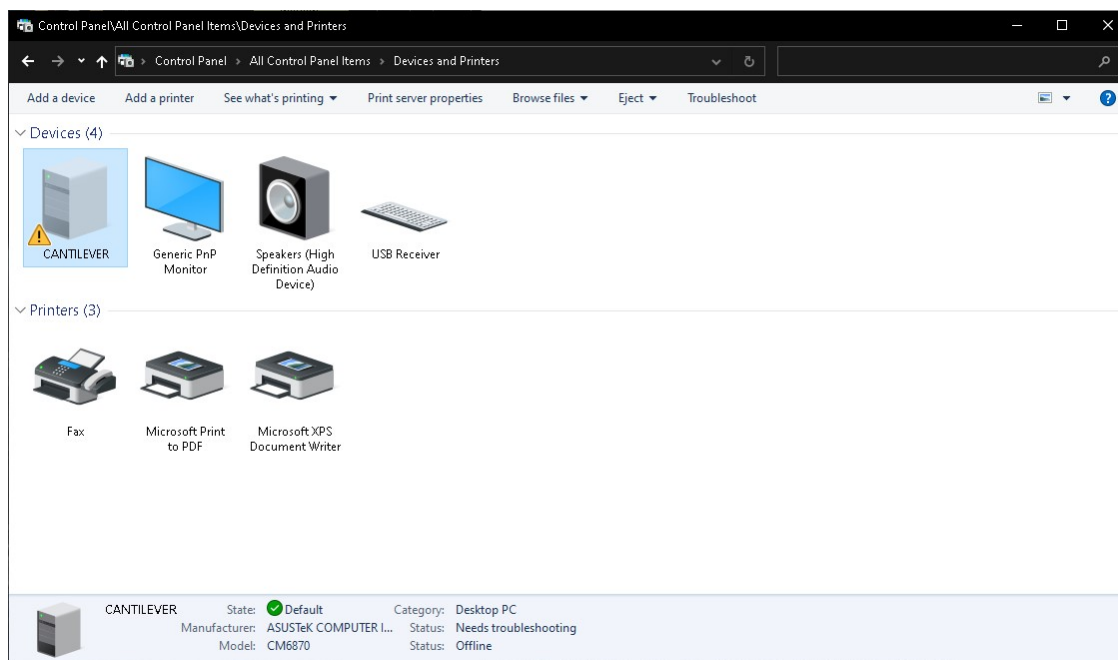
⁹ No practical way. I am leaving it as is so folks don't get the impression it's easy. In most cases the data is irretrievable. I can't do it but know folks who can.

Default Programs

Windows has a bad habit of resetting user choices to the defaults after updates in Windows 10. If the user complains that things don't look the same or they can't get to something any more try looking at the default programs, although this opens up in the settings app.

Device Manager

This is where you will go to resolve driver conflicts. It is also accessed from the computer management console.



Devices and Printers

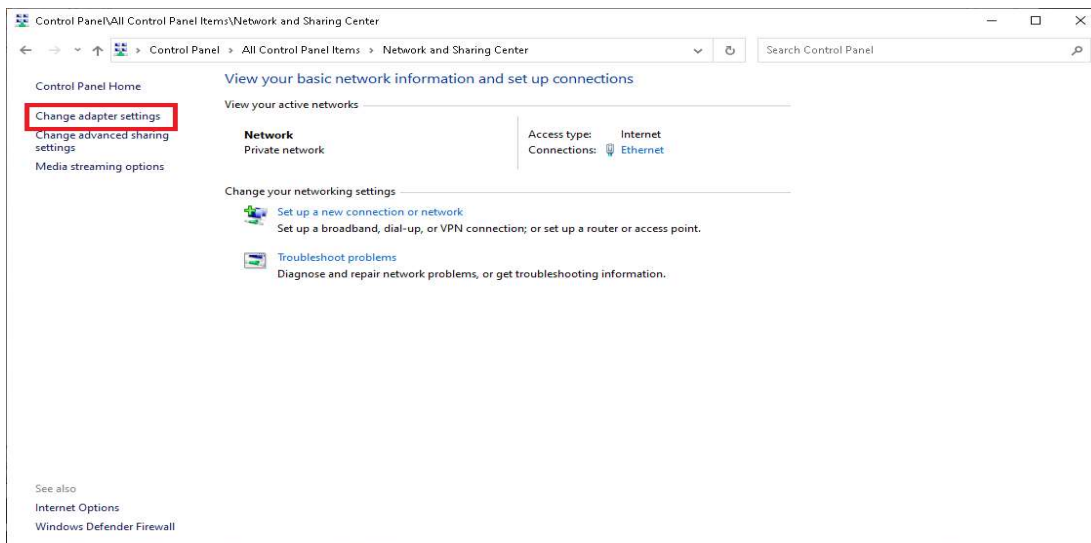
This is one way to add a printer, either by automatic detection by Windows, which has gotten much better, or manually. As you can see from the screenshot (my computer cantilever is working fine and online), Windows doesn't always get it right on the troubleshooting, which is why you should not rely on it and do your own manually. I like writing scripts that do a few things, and then run those and reboot when working in a timed environment.

File Explorer Options

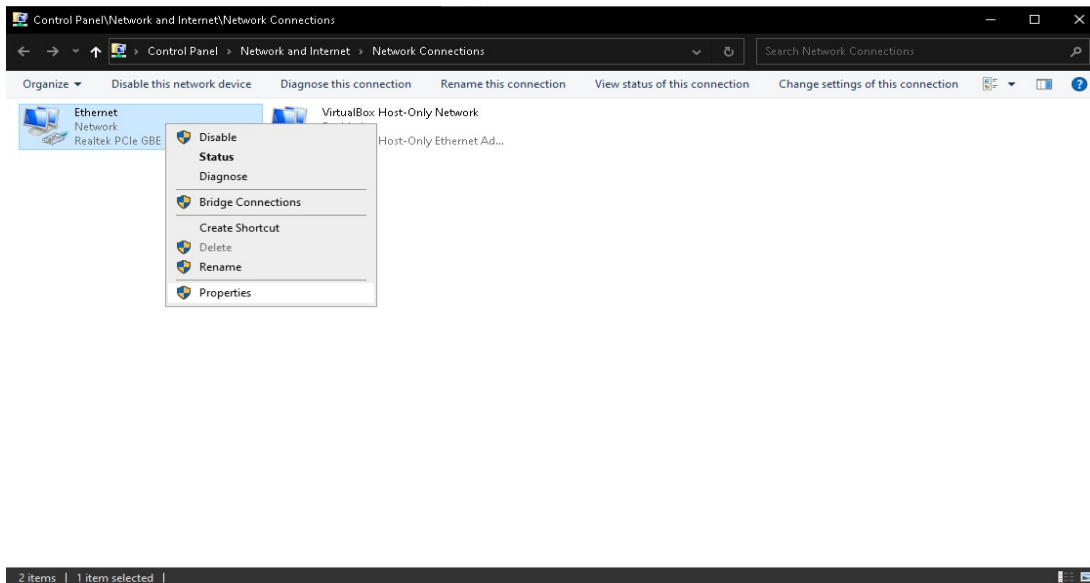
For added security you want to go to the View tab and ensure the radio button for 'Show hidden files, folders, and drives' is filled in and the box for 'Hide extensions for known file types' is unchecked. Both of these are not the default. The reason for showing the file extensions is so that malicious programs can't hide their origins easily. I like to show hidden files so that malicious files can't hide in plain sight by marking themselves hidden.

Network and Sharing Center

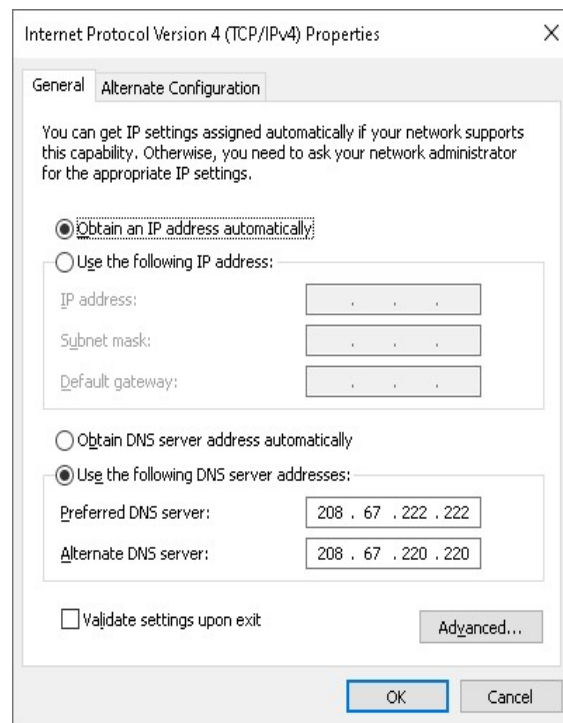
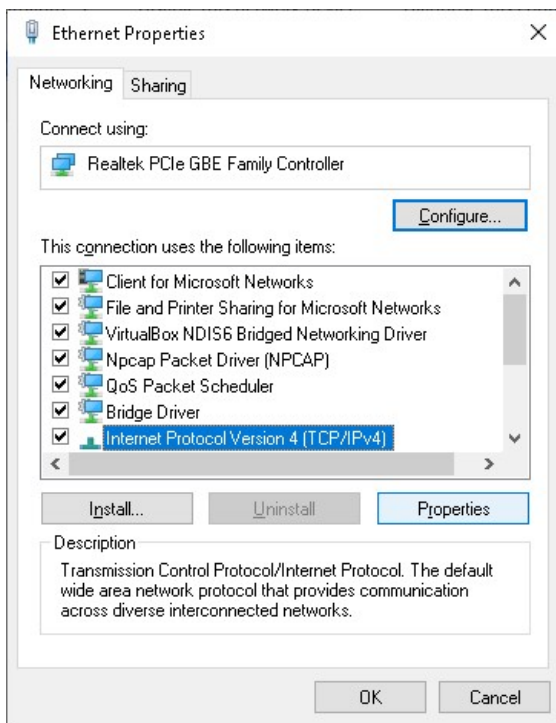
The network and sharing center will become very familiar to you. It is vital for troubleshooting internet issues. You can accomplish the same things in the command line or with powershell but most people find it easier with a GUI, and if you don't use either of those often or have it scripted, you will definitely use the GUI. Looking back on my experience, it is funny to me now that when troubleshooting Windows machines for a user I rely on the GUI more, but on Linux boxes I use the command line more. I definitely found scripting immensely helpful doing admin work, but for regular troubleshooting remotely and deskside, it was GUI almost exclusively.



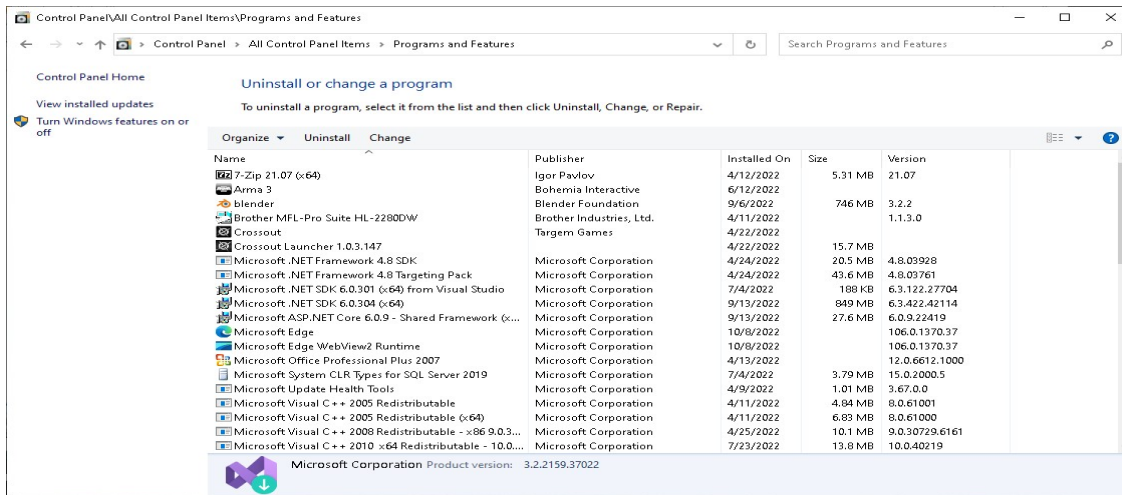
Click on 'Change adapter settings', then right click on the connection you're working on, in this case Ethernet, and select 'Properties' from the box that appears.



Highlight TCP/IPv4 and click on the 'Properties' button, and in the box that pops up you can enter your IP, internet protocol, address manually as well as your DNS, domain name service, servers. You need to click on the radio button to allow changes to be made.



If you have the radio button marked for 'Obtain an IP address automatically' like shown in the picture, that means you are using DHCP, dynamic host control protocol, which is how the router assigns IP addresses for NAT, network address translation. All of this will be explained further in the network section, but it made sense to put it in with the other control panel items for familiarization. Just know that you need it to do this for internet access right now.



Programs and features

The shortcut is `appwiz.cpl`, is how you add and remove programs and Windows features. When troubleshooting adware or spyware issues, open this panel and sort it by date installed to remove anything that may have been installed by the malware you are removing that would have given access back.

Sound

This is where you go for troubleshooting audio issues.

Storage Spaces

If you click storage spaces, it starts a wizard which guides you through setting up a RAID (redundant array of independent discs), a type of fault tolerant storage system that uses multiple hard drives.

Troubleshooting

You can use the troubleshooting consoles but they generally don't do any better than I do manually.

Users Accounts

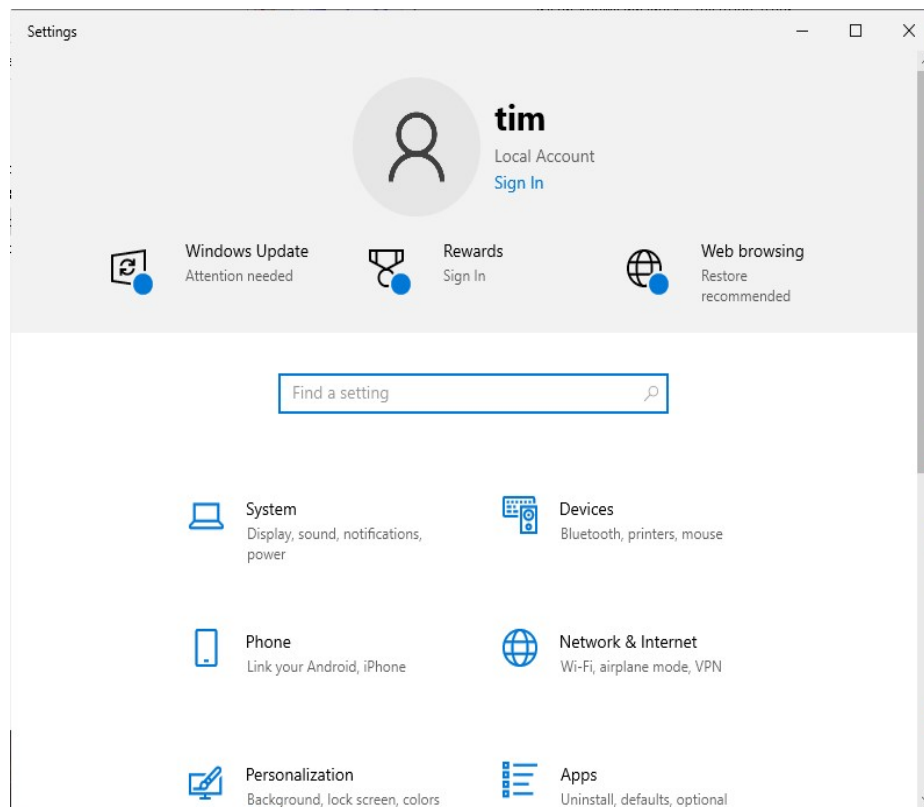
Most user account functions are handled in the settings app.

Windows Defender Firewall

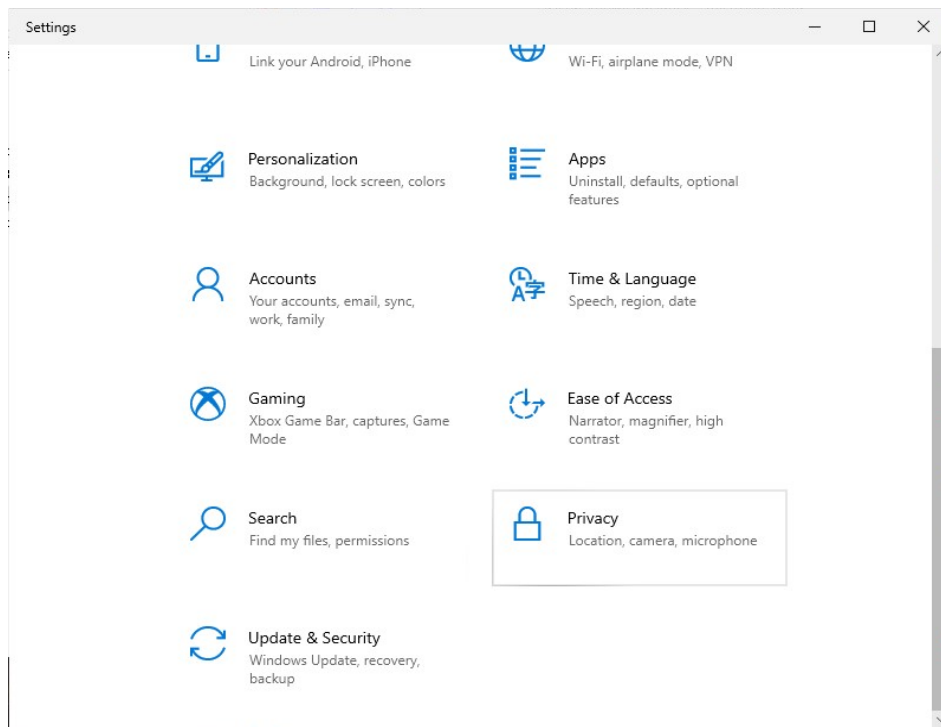
The firewall is handled more efficiently in the admin tools mentioned previously.

Settings

By now, you probably know how to get to the settings app, but in case you don't, either hit the Windows key and start typing, or hold the Windows key down and hit the letter 'R' to get the run box up and type 'settings' then hit enter.



We will cover the parts of the settings app that you will be using most often for troubleshooting and system configuration. You will find that some of the items are duplicates of the control panel, while some are only available here, and some are only available on the control panel. The important thing to remember is develop your troubleshooting techniques and stick to them so that you can repeat the results reliably. As you gain experience, you will get quicker. The most important thing as you are learning is learning the methods. Speed will come naturally as you familiarize yourself with your techniques.



System

Display is where you can change display settings such as color, screen resolution, size of text, and the multiple display configuration. For security purposes you should disable notifications on the lock screen. It is possible for criminals to obtain information about the system which can be used to gain unauthorized access. I consider it a best practice to disable almost all notifications from Windows, mainly to stop annoying messages rather than anything security related. Windows 10 makes it almost impossible to control your computer like you used to be able to, and it is getting worse in 11, so I don't expect it to ever get better. For security purposes you want to disable remote desktop. This can be enabled by the user as needed easily enough that you don't need to leave this avenue of attack open all the time.

Network & Internet

Another way to get to the network and sharing center and some of the network related settings which are similar to the control panel method of access. If you need anything network related and can't find it from the control panel, check here.

Personalization

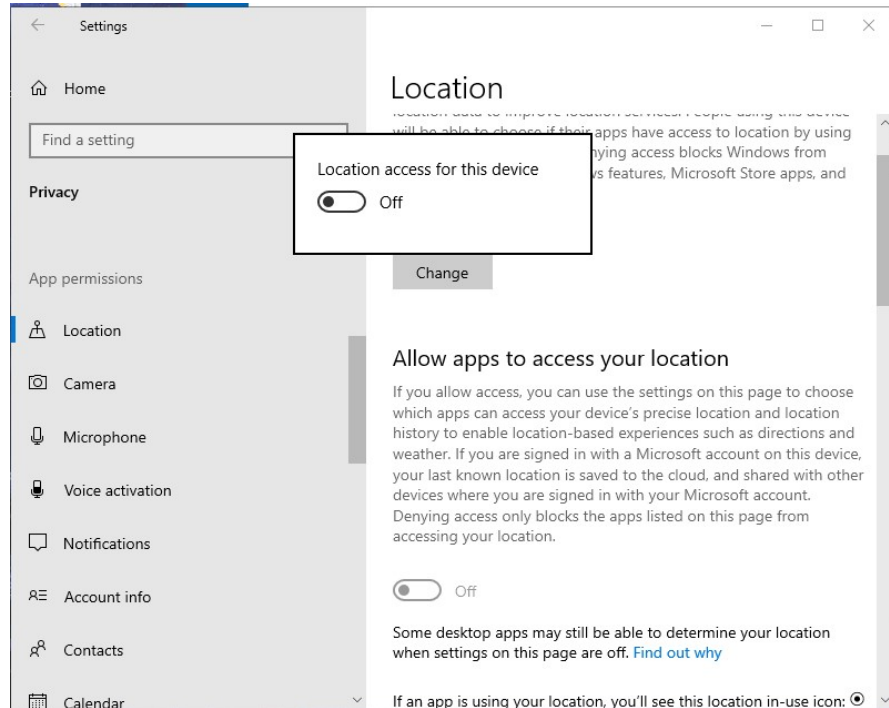
This is where you go to personalize the user experience. Things like desktop background, light and dark themes, what you see on the start menu, and how the taskbar is configured are here.

Ease of Access

Windows has several configuration settings for users who need additional assistance navigating a computer, such as screen narrator, magnifier, high contrast settings, closed captions and more.

Privacy

This one is important. You need to turn off access to everything listed under 'App permissions' on the left side by clicking on them one at a time and clicking the button that says change. When the box pops up, make sure it is turned off. This prevents malware from accessing information about the system and helps prevent infections. The picture shows the location access turned off. You can see by the scroll bar how many things need to be turned off.

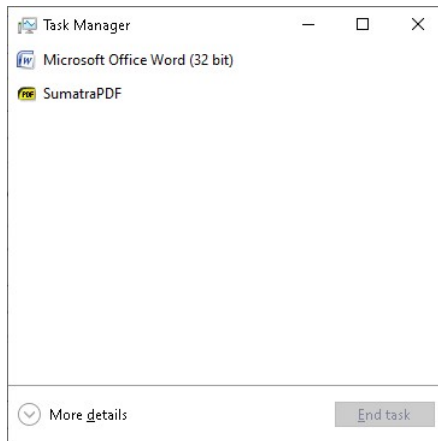


Update & Security

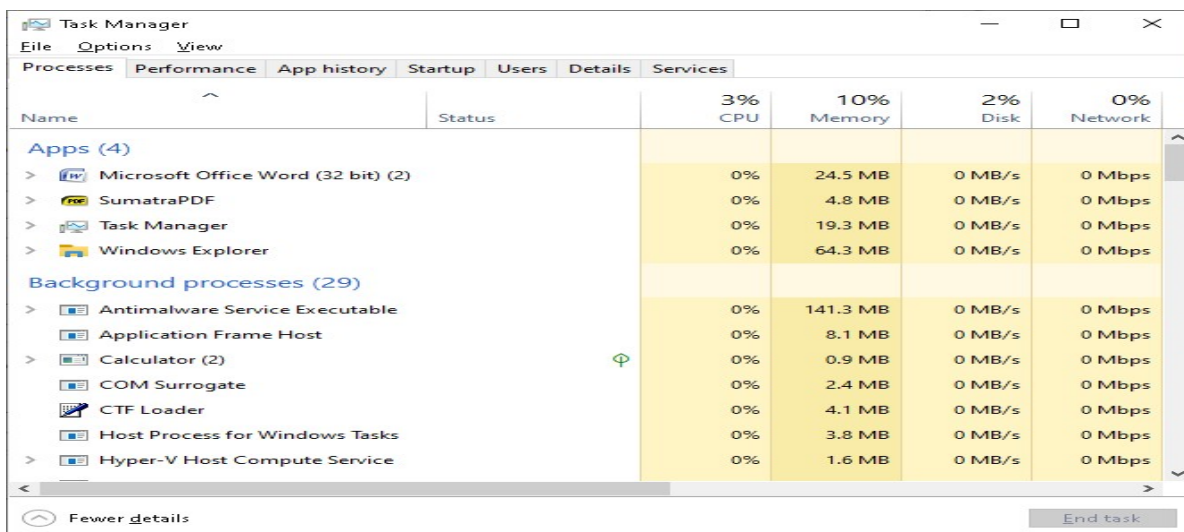
This is where you go to change the update hours, setup recovery options, activate windows, and set developer options. Prior to Windows 10 you could choose not to install updates. Now you can pause them for seven days. I figured out a way to stop Windows updates in Windows 10, but it is not a good idea and if I write about it Microsoft may fix the bug I'm using to do it, so I'm keeping that to myself.

Task Manager

This is one of the times where using the mouse is quicker than the keyboard. It doesn't happen often though. Right click on the taskbar and select *Task Manager* from the box that appears. This brings the task manager up with the default view, which is pretty basic but will allow you to kill a program that is stuck.



If you click on the drop down beside *More details*, you get a few more options.



As you can see in the picture, there are some useful things for troubleshooting here. The performance tab provides a view of the system resources and has a lot of good information for troubleshooting. App history shows the use of resources per application. It is a good place to look for the cause of excessive resource usage.

The startup tab has a list of all of the applications and processes that you want to start automatically when Windows starts. I like to disable all startup items so it doesn't impact the speed of startup as much. Everything is set to start when you want it to, and with the speed of modern computers, it doesn't take that long for the service to start on demand. When you add a startup item, it makes a registry entry in the run key. In fact, every configuration or customization you make creates a registry entry, since the registry is a database. More on all of that later.

The users tab shows which users are logged into the system, what services are have been started by that user, and what resources are being used by each of them. The services tab shows all of the services on the system and what state of each of them are in. The details tab shows details of the processes from the first tab.

File Structure

When it comes to operating systems, it's either Windows (DOS based, disk operating system) or Unix. Linux is Unix like, as is Apple, which is based on BSD, Berkley software distribution, an operating system very similar to the Linux kernel. With Apple and Linux the file structure is very similar, and most of the commands are interchangeable. Windows and Unix like systems share some similarities and a few commands are interchangeable but that's it. They all have their own file system format though. Windows uses FAT32 and now NTFS, Apple uses HFS+ and APFS, while Linux uses EXT mainly (EXT 2, 3, and 4) and also JFS, Reiser+, and others.

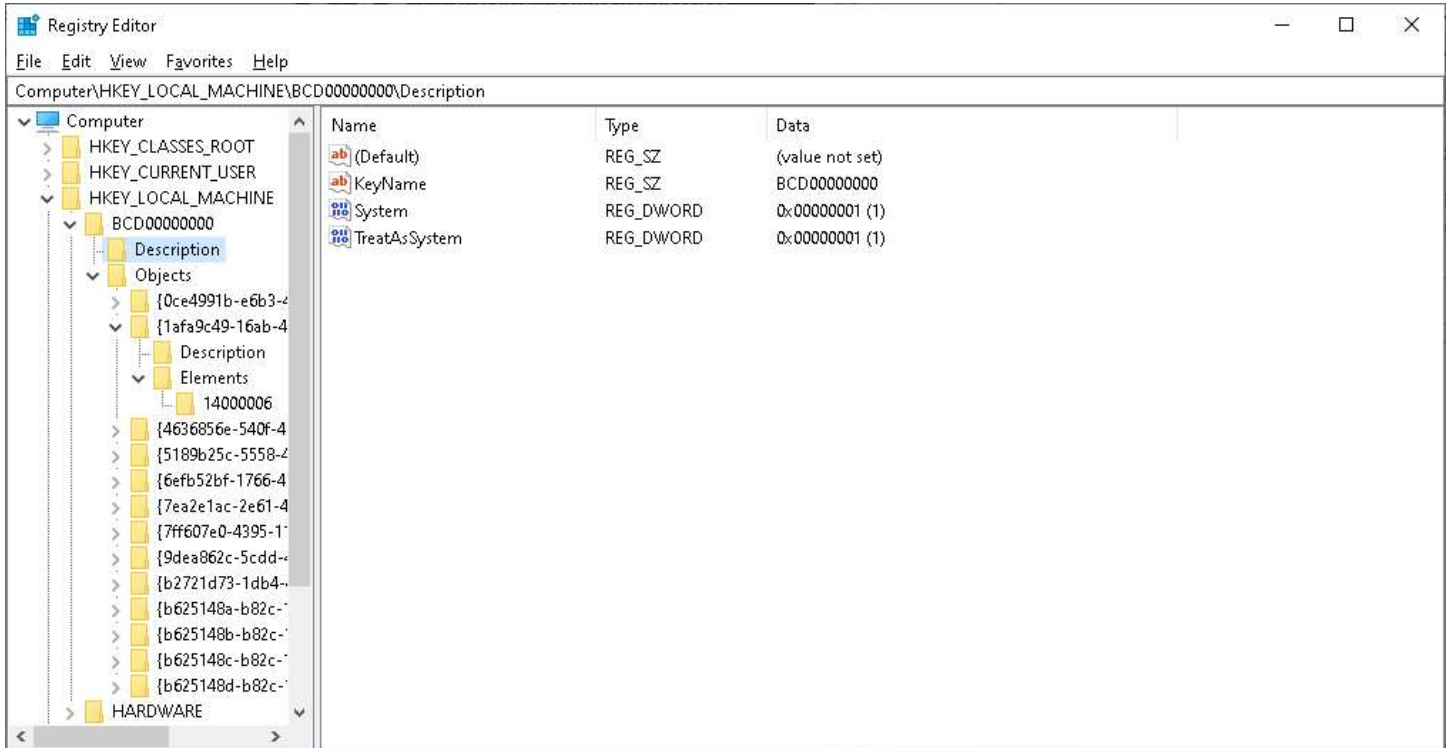
FAT stands for file allocation table, NTFS is new technology file system, HFS is hierarchical file system, APFS is Apple file system, EXT is extended file system, and JFS is journaling file system. All of these are different methods of keeping track of your data. Some can read others, but not write to them, while others cannot read the rest at all. Some of them can read and write to most of them, but I don't think any of them read and write to all of them.

When typing in the command line in Windows, the file path is written like this: C:\Users\tim\Desktop, while in a Unix like terminal it is written like this: /home/tim/Desktop. Notice the direction of the slashes; Windows uses back slashes while Unix like systems use forward slashes. The slash is named for the direction the top is pointing relative to the direction of writing. In the example commands above, C: in Windows refers to C drive, while the first '/' on a Unix like system is the root of the file system. Linux administrators are called root, probably because of this.

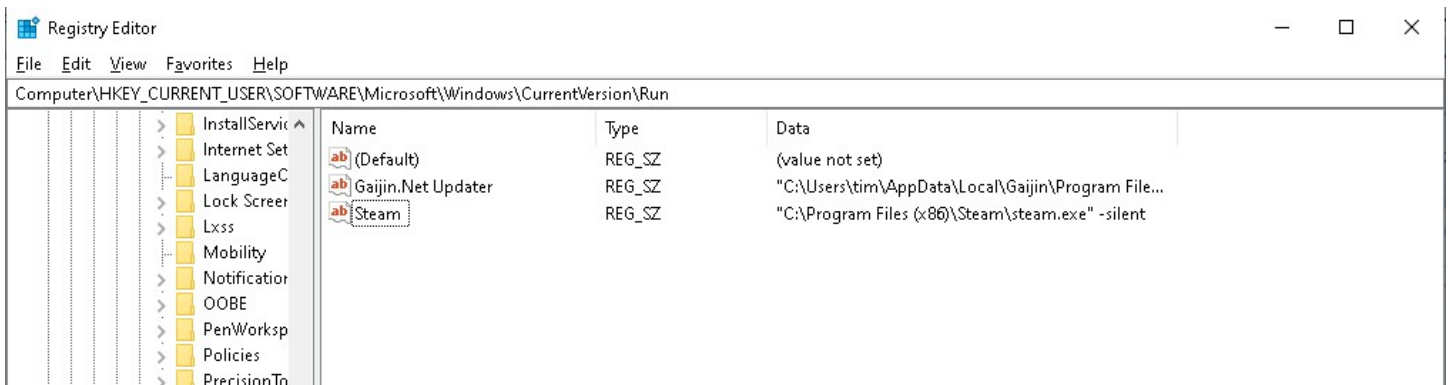
In Windows, it doesn't matter what case the letters are. Filename, filename, and FILENAME are all the same file, while on a Unix like system they refer to three different files. There are some other differences such as calling the command line from Windows a terminal on a Unix like system and most of the commands are different. Just keep in mind that all of the differences end up with the same ones and zeroes, just in different orders, but they mostly do the same job.

Windows

The registry in Windows is a database. It has what is called a tree structure because when you draw it in a picture it resembles a tree. In the following picture, the Computer is the bottom, and each arrow is a branch, with branches going out from branches.

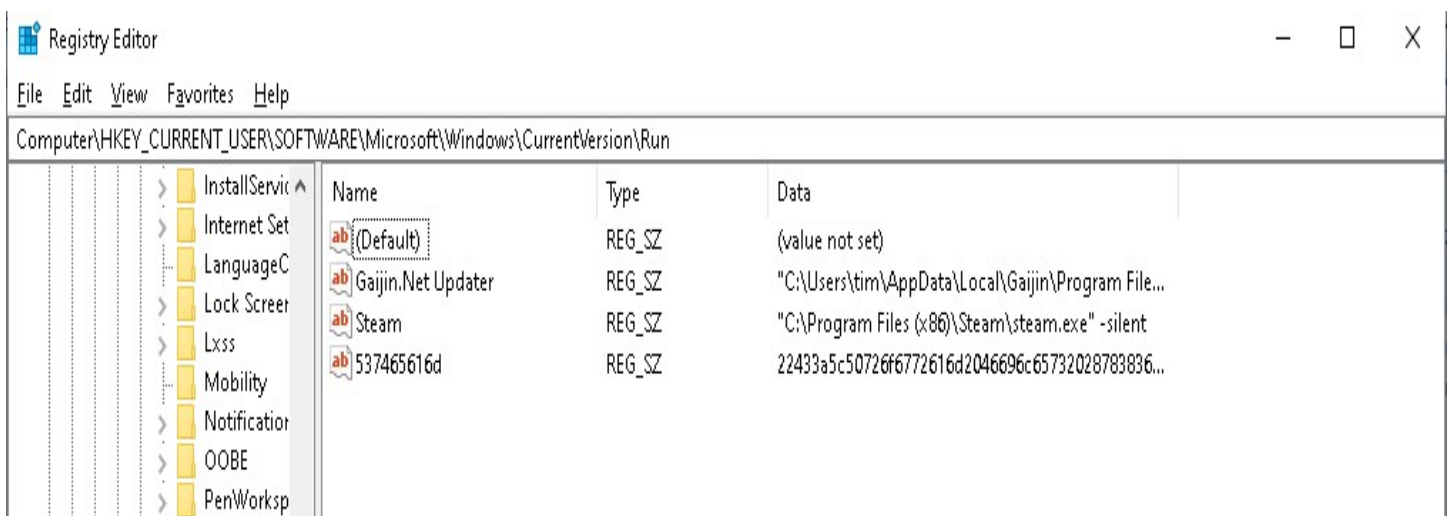


The registry contains all of the information Windows needs to operate. It is *hierarchical*, meaning that the permissions are inherited from the upper branch to the ones under it. The folders in the picture are called *keys* and the folders to their right are their *subkeys*. The first line of folders are keys; HKEY_CLASSES_ROOT, HKEY_CURRENT_USER, etc, and the folders to the right of them are subkeys. These subkeys can have subkeys or values, which is data, or both.



In this picture, notice the full path, HKEY_CURRENT_USER\SOFTWARE\Microsoft\Windows\CurrentVersion\Run. This subkey has some values inside. Every time the computer is started, one of the things that happens is this subkey is checked and if there is any data inside, it is processed. In this example from my computer, you can see there are two programs that execute, and they are both enclosed in double quotes. This tells Windows to interpret the data as a string. It is the same as if you opened up a command prompt and typed those letters in and hit enter. You can't see all of the first one, but the second also has a -silent after the string. This is called a flag, and in this case tells the program to run in the background, or silent. We'll learn more about flags in the next section.

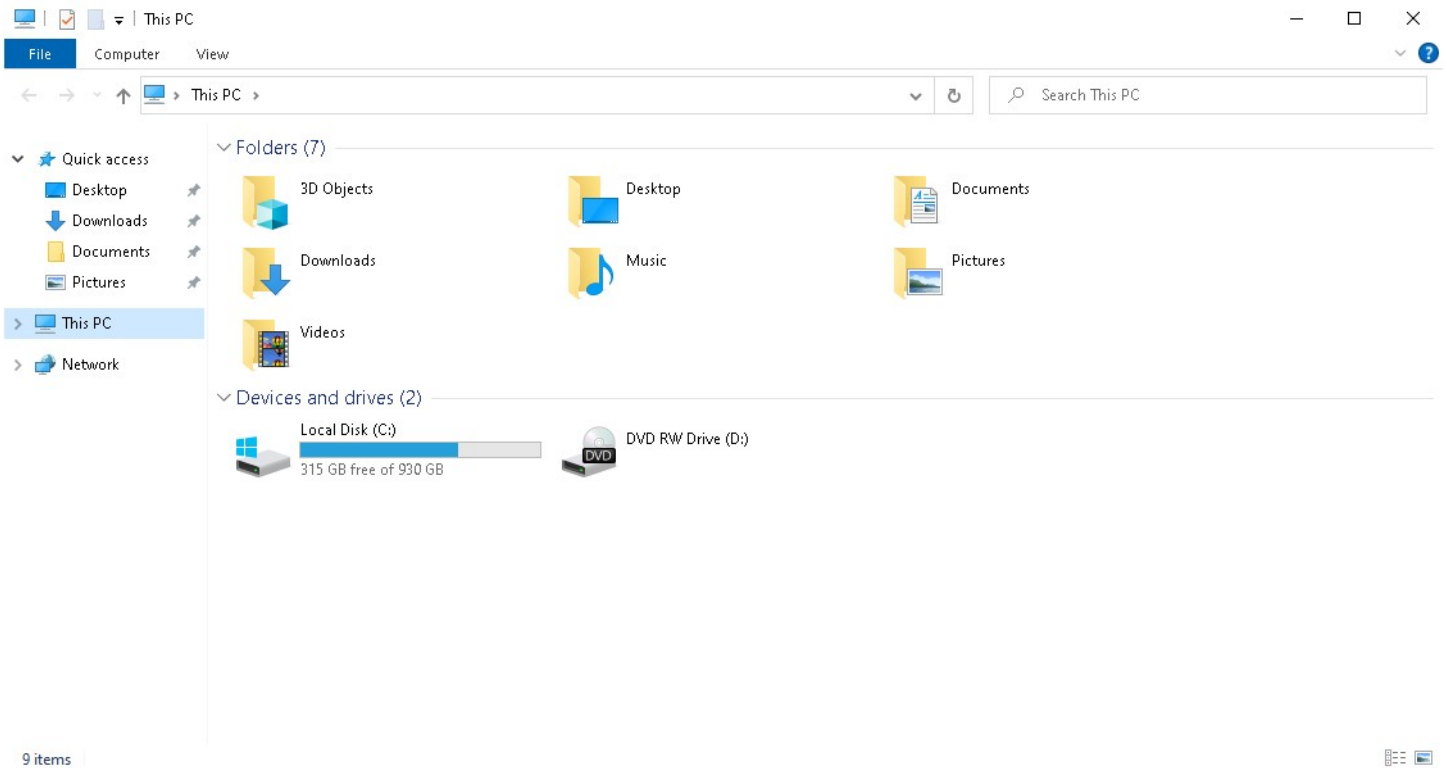
You will learn to check this subkey and a few others as a source of malware. That is because the computer doesn't care what lives here, it runs it. In this case, it was a legitimate example, but in the next picture, who knows what it is?



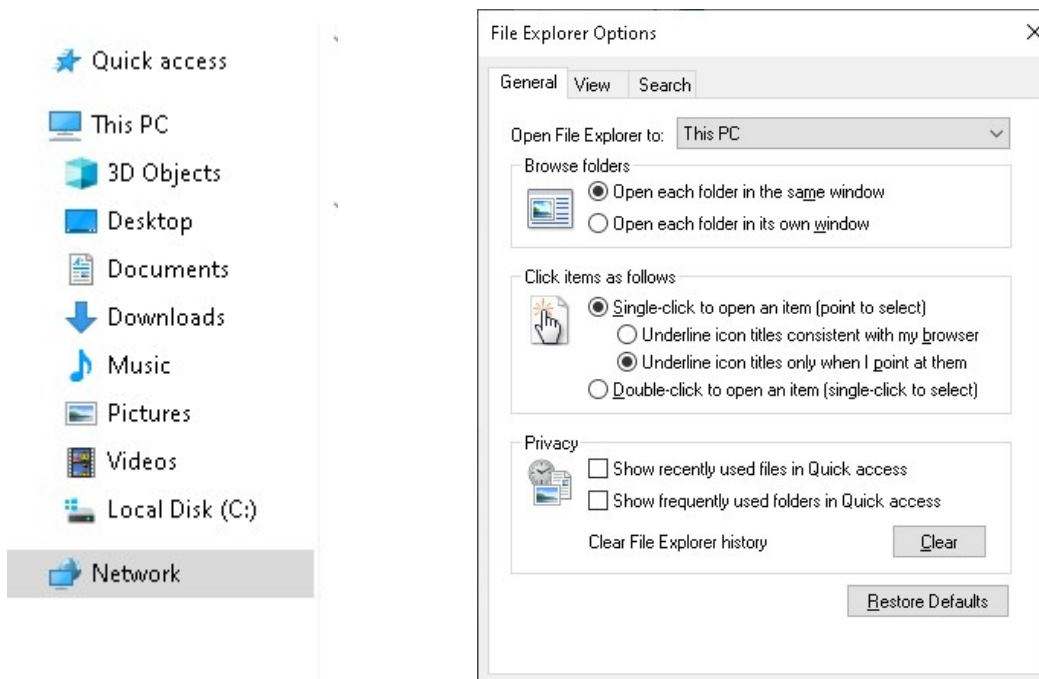
If you copied those values into a hexadecimal translator you would find that it is a copy of the Steam value above. The computer doesn't care that you can't read it; it takes the value and runs it as if it were a command, just like all of them. This is how malware can hide here. If you recall the startup tab of the task manager, this is the place in the registry where those entries point to.

This PC

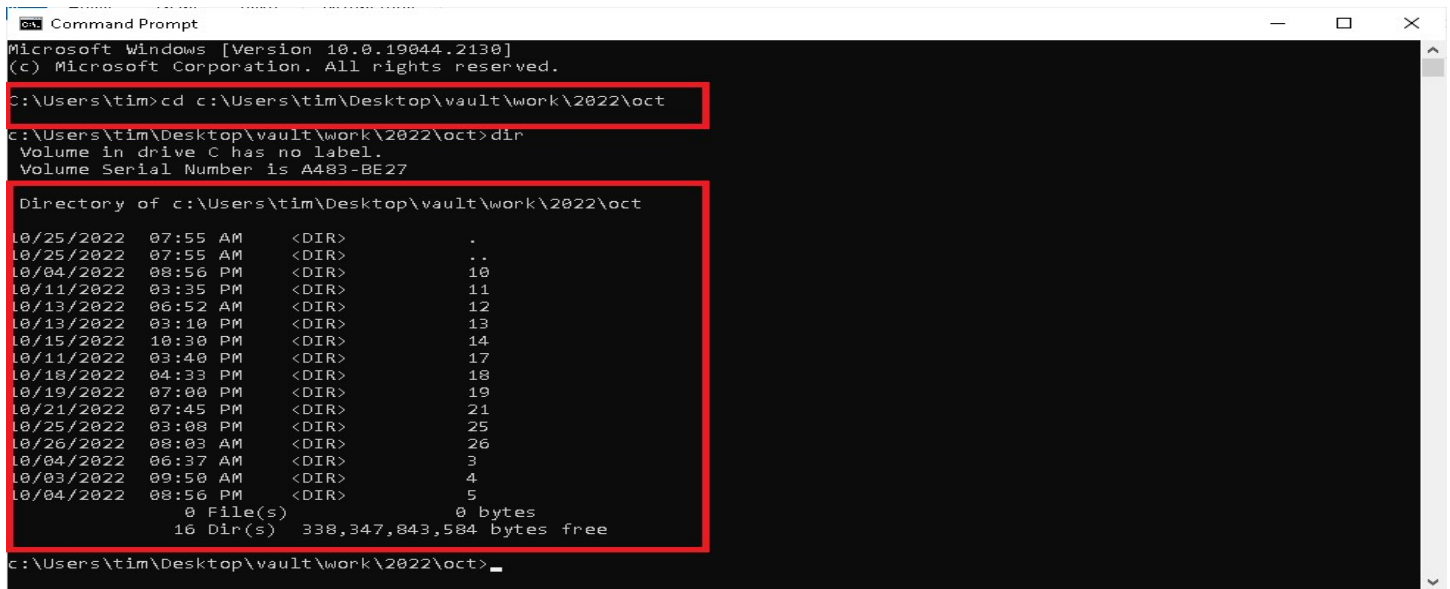
Even though Windows uses a tree structure, it uses folders for the icons, and that is how I will be referring to it now for simplicity. If you click on the folder icon in the taskbar, it opens up like so:



One of the changes I like to make is to set This PC to be expanded instead of Quick access in the first tab of the file explorer options from the control panel, as shown here:



Recall that I mentioned the file path earlier, and that C: was C drive. In the following two pictures, they both show the same directory, but one is using the GUI and the other is using the command prompt, known as a terminal in Linux. Note the address is the same for both pictures and they have the same folders.



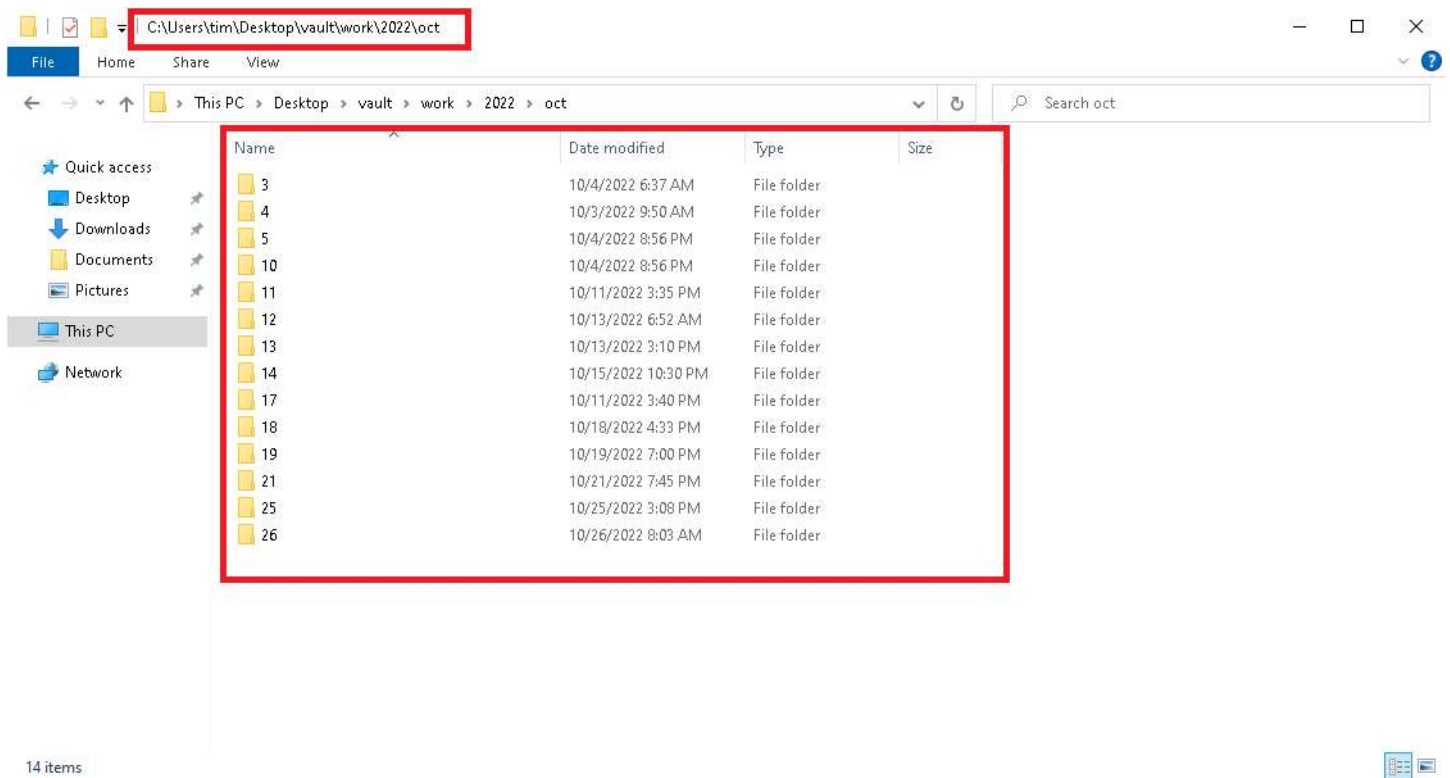
```
Microsoft Windows [Version 10.0.19044.2130]
(c) Microsoft Corporation. All rights reserved.

C:\Users\tim>cd c:\Users\tim\Desktop\vault\work\2022\oct
C:\Users\tim\Desktop\vault\work\2022\oct>dir
Volume in drive C has no label.
Volume Serial Number is A483-BE27

Directory of c:\Users\tim\Desktop\vault\work\2022\oct

10/25/2022  07:55 AM    <DIR>      .
10/25/2022  07:55 AM    <DIR>      ..
10/04/2022  08:56 PM    <DIR>      10
10/11/2022  03:35 PM    <DIR>      11
10/13/2022  06:52 AM    <DIR>      12
10/13/2022  03:10 PM    <DIR>      13
10/15/2022  10:30 PM    <DIR>      14
10/11/2022  03:40 PM    <DIR>      17
10/18/2022  04:33 PM    <DIR>      18
10/19/2022  07:00 PM    <DIR>      19
10/21/2022  07:45 PM    <DIR>      21
10/25/2022  03:08 PM    <DIR>      25
10/26/2022  08:03 AM    <DIR>      26
10/04/2022  06:37 AM    <DIR>      3
10/03/2022  09:50 AM    <DIR>      4
10/04/2022  08:56 PM    <DIR>      5
0 File(s)          0 bytes
16 Dir(s)         338,347,843,584 bytes free

C:\Users\tim\Desktop\vault\work\2022\oct>
```



This is a good example of how there is always more than one way to do something. On the command prompt in the picture, you could type `cd c:\users\tim` and hit enter to change to the tim directory. If you were using the second picture, you could click the mouse just behind the oct above the large red box and type `c:\users\tim` and hit enter, it also takes

you to the tim directory. Or you could click through, what we call it when we just click on the folders individually, also known as "drilling down". You can use wildcards in file paths just as you can in the terminal. %systemroot\System32 is the same as C:\Windows\System32. That is how you go to the root of the operating system in case it is installed somewhere other than C:. You can also type %appdata% and hit enter to go to the roaming profile of the AppData folder for the current user.

This also illustrates the concept of *nested folders*. The folders to the right are nested inside the one to the left. C drive is the hard drive and is called the *root* of the drive. I like to use them as an easy way to back up my computer. I make a folder on the desktop and name it something you store things in. Now it is vault, but I have used house, cave, place, and other things. Inside this folder I make a folder for each type of thing I need like work, music, pictures, reference, software, etc. If I need to further separate things I just make folders nested inside them.

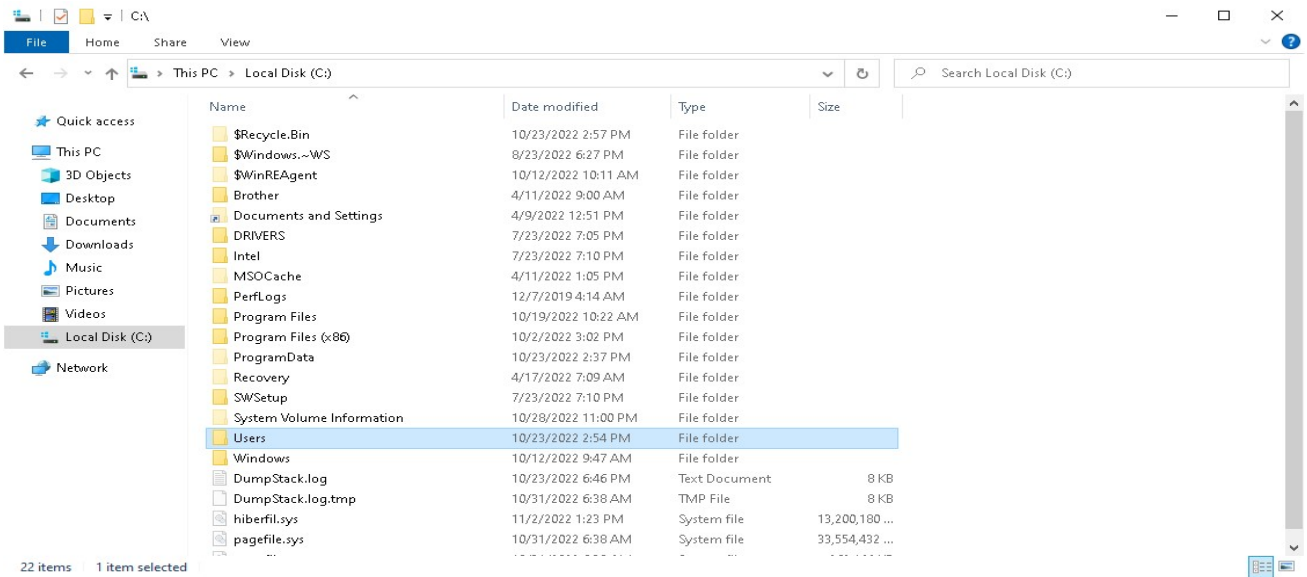
You have to watch out because although there is no real limit to how many folders are nested inside others (4,294,967,295 files per volume*) there is a 260 character limit on filenames. While this isn't that small, you can see that with long folder and file names and a lot of nested folders it could go over the limit. If you find this limit preventing you from accessing files, start at the leftmost folder and rename them to one letter names. Eventually you will free up enough space to open them.

The way it makes the backup really easy is that after you export your bookmarks, you put that in the backup folder on the desktop and then copy that one folder to the external device. You don't need to worry if you got all the latest pictures or that paper you were working on because everything is stored in that one place. There are some programs that store data that needs backed up, but you can change the configuration in most of them to save it to your backup folder instead.

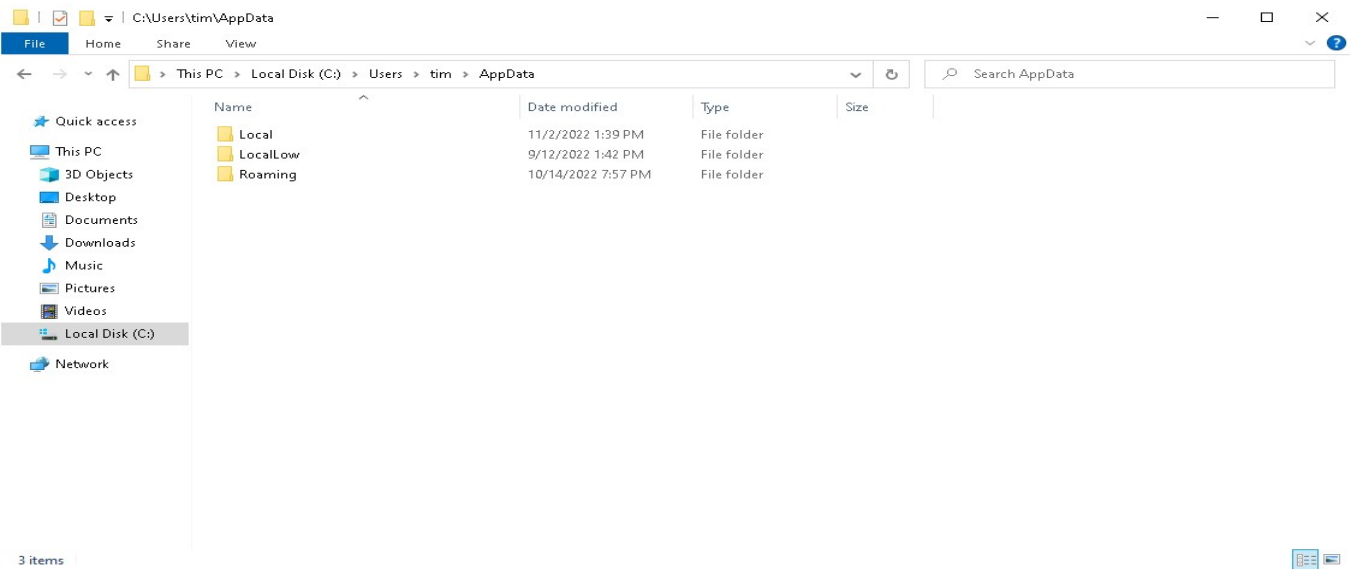
Standard Folders

Every Windows 10 installation has the same standard folders in what is known as a *hierarchy*. When programs are installed in Windows, they use either the *Program Files* or *Program Files (x86)*. The (x86) folder is for 32 bit programs, while the other is for 64 bit programs. The files that are slightly lighter in color are hidden. You would think that drivers would be in the DRIVERS folder, but you'd be wrong. They are located in C:\Windows\System32\DriverStore.

* [https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2003/cc781134\(v=ws.10\)?redirectedfrom=MSDN](https://learn.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2003/cc781134(v=ws.10)?redirectedfrom=MSDN)



The other folders you should be aware of are the AppData folders; typing `%appdata%` in the address bar takes you there. These are where the temporary files are kept along with where most applications store data, hence the name. There are three folders inside, Local, LocalLow, and Roaming. They are system files created during installation. The Local folder has the files that are local to the computer and don't leave when the profile is copied or the roaming Desktop is used. LocalLow is also local, except with lower privileges than Local, while Roaming has the data that is used when the roaming profile is used or when the profile is copied.



Typically, you will look here for data when you need to recover data from a crashed hard drive, before reinstalling windows, or changing out hard drives. Always ask your clients if they use banking software such as Quicken or Quickbooks. You will need to retrieve the database and do a backup so you don't lose their financial data. More companies are moving

their records online so it is less of an issue if you forget, but you need to be aware and make sure to ask before proceeding.

Command Prompt

The Windows command prompt is the command line interface, or CLI, that allows you more fine grained control over the operating system. When personal computers first came out in the 1980's, this was all you had. It was all DOS based or BASIC, which was what mine ran. We could draw primitive graphics, but the GUI didn't come along until the '90's. You can still do all sorts of things in the CLI; you can even write programs called batch files which we will spend time on soon.



```
Administrator: Command Prompt
Microsoft Windows [Version 10.0.19044.2130]
(c) Microsoft Corporation. All rights reserved.
C:\Windows\system32>
```

Notice at the top it says "Administrator Command Prompt". This is known as an elevated command prompt and you can see at the cursor the file path says, "C:\Windows\system32". This is where the command prompt originated, the system32 folder of Windows, which along with the SysWOW64 folder is where the system files live. Look back at the command prompt window before this one to see what the file path is. We call it elevated because it requires UAC elevation. To open an elevated command prompt, hit the Windows key and type 'cmd', but instead of hitting enter, click on 'Run as Administrator' to the right of the command prompt (you may have to right click to get that choice). This will trigger the UAC box to have a password entered or the yes button pressed, depending on your UAC settings.

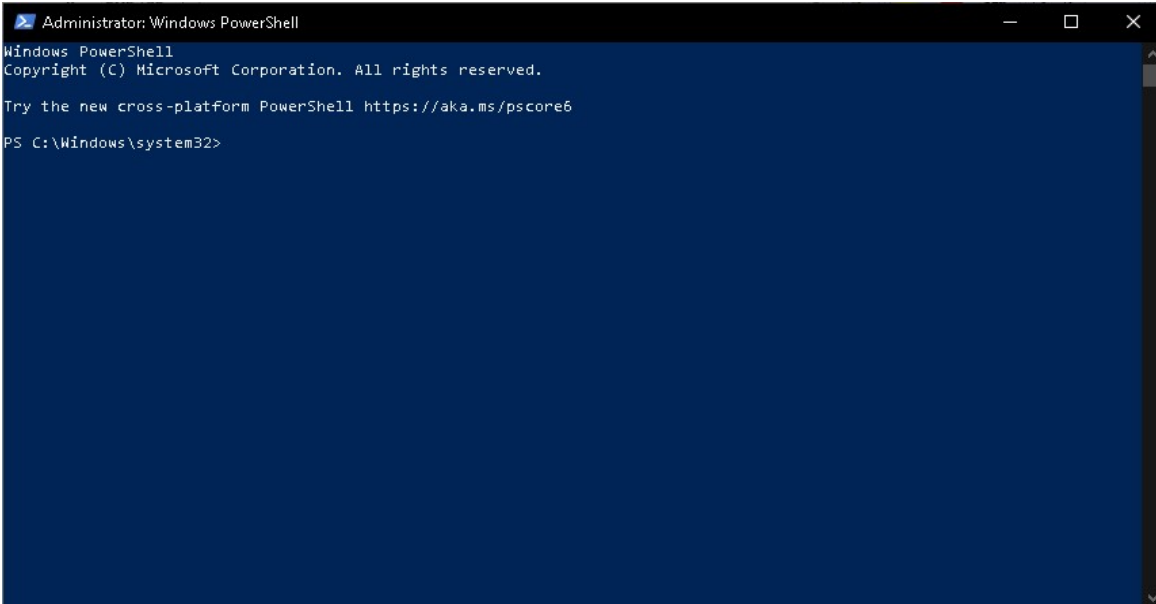
By default the cursor has the file path displayed. In order to navigate, you type *cd filepath-you-want-to-go-to* and hit enter. For example, *c:\users\tim\documents* takes you to the documents folder for the user tim. To change drives, type the drive letter and hit enter. If you want to see the contents of the directory you are in, type 'dir' and hit enter. You can see the results of these in the earlier picture.

It is important to know how to navigate using the CLI for data recovery and system administration. When you can't get a Windows system to boot,

you may be able to retrieve the data if you boot from a Windows install disk, and then choose the option to cancel the installation and open a command prompt. This allows you to use an external hard drive to transfer the data, but you need to use the `scp`, or secure copy command, and you need to know the proper flags to use. You can't just drag and drop the files into the folder if the only thing you can use is the command prompt. We will cover how to get around this in the scripting section.

PowerShell

PowerShell is similar to the command prompt, but is more versatile. It was developed in 2006 when Microsoft renamed one of their scripting engines. The command prompt was integral to Windows and ran everything but did not have a fine grained control of the GUI or all of the configurations. PowerShell changed all of that. In 2016 Microsoft open sourced PowerShell and made it cross platform also. There are hundreds of PowerShell modules available from Microsoft as well as 3d party modules. These allow for complete control of the system and automation of repetitive tasks. If you are installing 3d party modules, make sure they are from a trusted source, as malware can easily hide inside and evade detection. Like the command prompt, it also has an admin mode which is marked and accessed the same way.



```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Windows\system32>
```

The syntax, known as grammar in English or the way we write commands, is not interchangeable with the command prompt. If you type `get-computerinfo` and hit enter you will get a whole lot of information. Navigation works the same way, in the example above if you typed `cd ..` and hit enter it would take you up one directory to `C:\Windows`. We will get more familiar with powershell and command prompt syntax in the script section.

Scripting

When we talk about scripting, we mean writing small programs in whatever command line you are using, such as the Windows command prompt or PowerShell, or the Apples or Linux terminal, usually Bash or some derivative of a shell language. Scripts in Apples and Linux have a file extension of `.sh`, and Windows Powershell scripts have a `.ps1`. Command prompt scripts are known as batch files and end in `.bat`. We're going to start with those.

Batch Files

Batch files are scripts that run commands when you click on them. They can either be single complicated commands with a lot of switches or a series of commands. They can run silently in the background or with user interaction. Since they run with the permissions of the user logged in, if you need admin privileges, you need to right click on it and select *Run as Administrator*. I will also show you how to add a prompt for UAC so that it will ask for the permission.

Batch files start with `@echo off`. This stops the display from going to the terminal window. Open notepad and type `@echo off` then hit enter. Then type `ipconfig`, hit enter, then type `pause`. Save the file on the desktop and in the dropdown under the filename at the bottom, make sure to select *All Files (*.*)*, then name it `ping.bat`. When you click on it, it will run the command `ipconfig` and then pause, waiting on user input. The command prompt window will pop up and show the Windows IP configuration and at the bottom it will say, "Press any key to continue..." and when you press a key, it closes the window.

The next batch file I will show you is one I have been using for a while now. I call it *no-internet* and it needs admin privileges to do everything it needs to do to restore internet access. I used to just right click on it and select *Run as Administrator*, but with just a little bit of research, I found some code to ask for elevated privileges and added it to the beginning. Now when you click on it, it checks to see if it is running in an elevated command prompt and if it isn't it asks for permission, then once granted it runs the operation. You can copy the code inside the box and paste it into notepad. When you save it, make sure to change the file type as noted above. You should see two gears on the icon indicating it is a batch file now and not a text file.

```

@echo off

:: BatchGotAdmin

:-----

REM --> Check for permissions

>nul 2>&1 "%SYSTEMROOT%\system32\cacls.exe"
"%SYSTEMROOT%\system32\config\system"

REM --> If error flag set, we do not have admin.if
'%errorlevel%' NEQ '0' (

    echo Requesting administrative privileges...

    goto UACPrompt

) else ( goto gotAdmin )

:UACPrompt

    echo Set UAC = CreateObject^("Shell.Application"^) >
"%temp%\getadmin.vbs"

    echo UAC.ShellExecute "%~s0", "", "", "runas", 1 >>
"%temp%\getadmin.vbs"

    "%temp%\getadmin.vbs"

    exit /B

:gotAdmin

    if exist "%temp%\getadmin.vbs" ( del
"%temp%\getadmin.vbs" )

    pushd "%CD%"

:-----

ipconfig /release

ipconfig /renew

ipconfig /flushdns

ipconfig /registerdns

netsh dump

nbtstat -R

netsh int ip reset reset.log

netsh winsock reset

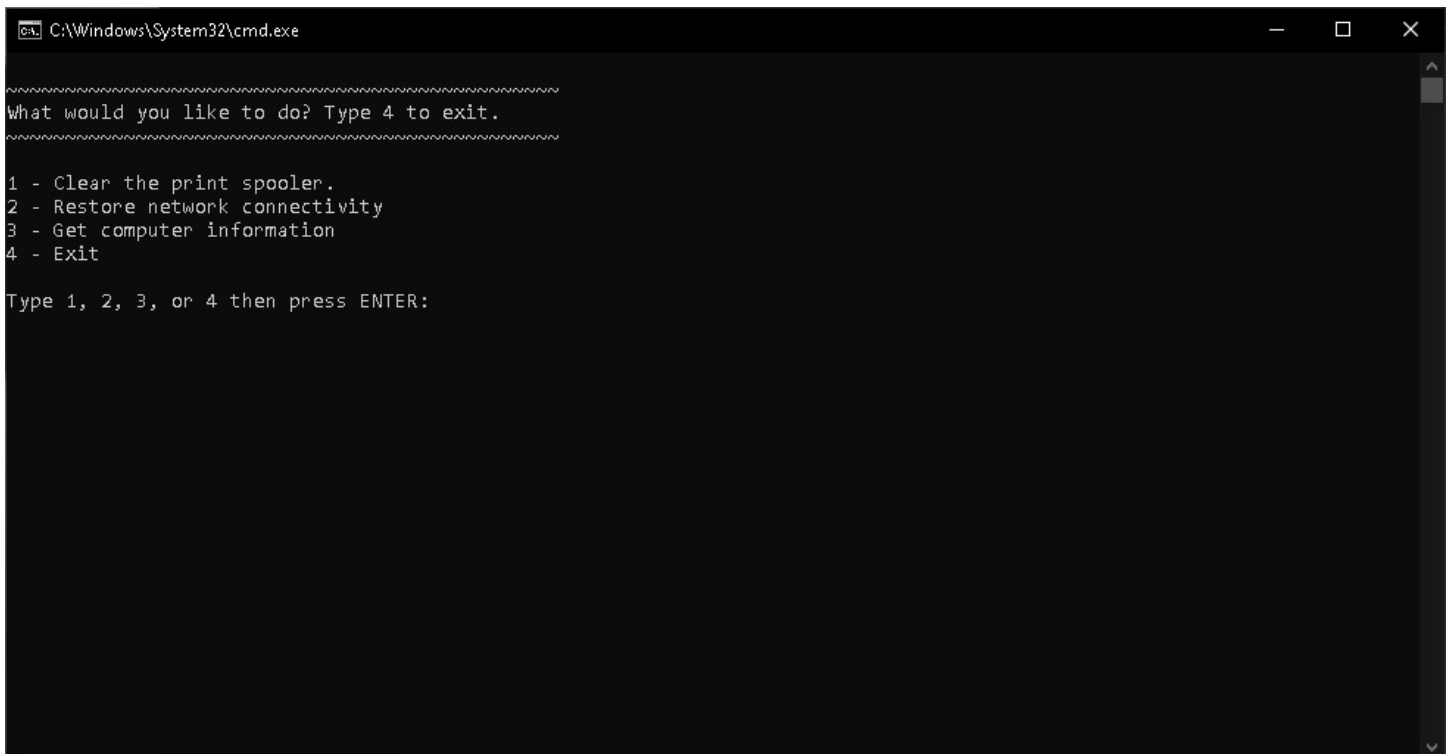
pause

```

Notice that the part between the lines (:-----) are what asks for the UAC elevation. Everything under that are commands that get executed to perform their individual functions to help restore network connectivity. We will go into what they do during the network section.

Now that you know how to make a batch file ask for elevated permissions, it's time to learn how to make a menu driven batch file. This will display a menu allowing you to choose from a selection of things. Notice in the previous batch file there were several places that had a colon (:), such as :UACPrompt and :gotAdmin. Those are called *anchors* and are there to specify specific sections of the program to run. When two colons are put together (::) it creates comments where you can describe what is going on. Comments are not treated like commands and you should use lots of them when you write programs. Coming back to them after years or looking at code someone else wrote can be time consuming and difficult when there are few comments.

You should recall that typing *@echo off* will stop Windows from displaying the commands being run next. When you type just *echo off*, it suppresses the next command. When you type *echo*, it displays whatever you type after that, and typing a dot after *echo* will skip a line on the display. For example, if you open a command prompt and type *echo the clouds are low*, then it will display, "the clouds are low" and a cursor under it, and if you type *echo.* then hit enter it will display a blank line and the cursor under it. For our menu, we start with *echo off*, then *cls* to clear the screen. Then we use what we learned about *echo* to set up our menu. When you get done, it should look similar to the picture below.



```
C:\Windows\System32\cmd.exe
What would you like to do? Type 4 to exit.
1 - Clear the print spooler.
2 - Restore network connectivity
3 - Get computer information
4 - Exit
Type 1, 2, 3, or 4 then press ENTER:
```


We used the echo command to display the text and a dot after it to put the empty lines in. We use set /p to take user input (/p) and set that as the variable (set). Then we read the variable and go to the appropriate section and execute the code there.

```
:menu

ECHO.

ECHO ~~~~~

ECHO What would you like to do? Type 4 to quit.

ECHO ~~~~~

ECHO.

ECHO 1 - Clear the print spooler.

ECHO 2 - Restore network connectivity

ECHO 3 - Get computer information

ECHO 4 - Quit

ECHO.

SET /P M=Type 1, 2, 3, or 4 then press ENTER:

IF %M%==1 GOTO print

IF %M%==2 GOTO network

IF %M%==3 GOTO compinfo

IF %M%==4 GOTO EOF

:print

@echo off

net stop spooler

cd c:\windows\system32\spool\printers

del /Q *.*

net start spooler

goto menu

:network

@echo off

ipconfig /release

ipconfig /renew

ipconfig /flushdns
```

```

ipconfig /registerdns

netsh dump

nbtstat -R

netsh int ip reset reset.log

netsh winsock reset

echo You need to restart now to finish configuration changes..

::set /p answer=Restart now?

::if %answer%="y" shutdown /r /t 0 else

::if %answer%="yes" shutdown /r /t 0 else

pause

goto menu

:compinfo

@echo off

REM set variables

set computer=

set system=

set manufacturer=

set model=

set serialnumber=

set osname=

set sp=

set cstring=

set ustring=

set pstring=

FOR /F "tokens=2 delims=''" %%A in ('wmic %cstring% %ustring% %pstring%
OS Get csname /value') do SET computer=%%A

FOR /F "tokens=2 delims=''" %%A in ('wmic %cstring% %ustring% %pstring%
OS Get csname /value') do SET system=%%A

FOR /F "tokens=2 delims=''" %%A in ('wmic %cstring% %ustring% %pstring%
ComputerSystem Get Manufacturer /value') do SET manufacturer=%%A

FOR /F "tokens=2 delims=''" %%A in ('wmic %cstring% %ustring% %pstring%
ComputerSystem Get Model /value') do SET model=%%A

FOR /F "tokens=2 delims=''" %%A in ('wmic %cstring% %ustring% %pstring%
Bios Get SerialNumber /value') do SET serialnumber=%%A

FOR /F "tokens=2 delims=''" %%A in ('wmic %cstring% %ustring% %pstring%
os get Name /value') do SET osname=%%A

FOR /F "tokens=1 delims='|'" %%A in ("%osname%") do SET osname=%%A

FOR /F "tokens=2 delims=''" %%A in ('wmic %cstring% %ustring% %pstring%

```

```
echo done!

echo -----

echo System Name: %system%

echo Manufacturer: %manufacturer%

echo Model: %model%

echo Serial Number: %serialnumber%

echo Operating System: %osname%

echo Service Pack: %sp%

echo -----

REM Generate file

SET file="%~dp0%computer%.txt"

echo ----- > %file%

echo Details For %computer%: >> %file%

echo System Name: %system% >> %file%

echo Manufacturer: %manufacturer% >> %file%

echo Model: %model% >> %file%

echo Serial Number: %serialnumber% >> %file%

echo Operating System: %osname% >> %file%

echo Service Pack: %sp% >> %file%

echo ----- >> %file%

echo File created at %file%

pause

goto menu
```

You may have noticed that the network section is just the no-internet batch file. I copied three batch files I used for other things to show how easy it is to automate things. That's about enough for batch files, time to move on to PowerShell.

PowerShell

PowerShell is very different from the command prompt. It is a shell and scripting language. It was designed from the ground up for a different purpose and it is very complex. I am no expert in PS, so I am going to lean heavily on Microsoft documentation. Since PS could have its own book I won't cover it completely, but will provide links for reference and as always I encourage you to explore further. Microsoft has some great tutorials for learning PS that I used and highly recommend[∞].

As noted in the file structure section, PS has the same type of environment as the command prompt, the standard user interface and the administrator interface. If you run a PS script or command and it returns an error, look at what it says closely. Sometimes it is a permissions error meaning it requires elevated privileges but is running in a standard interface.

PS operates differently than most shells by acting on objects instead of executing commands from text and returning text. By treating everything as an object, it allows for faster use by *pipelining*, sending the output of one command to the input of another. Objects have *properties* and *methods*, what it is and what can be done to it. All objects that are the same have the same properties and methods. Since PS uses objects rather than text, it can work on the properties the same for everything. That's what makes the pipelining work faster; it doesn't have to convert from one format to another.

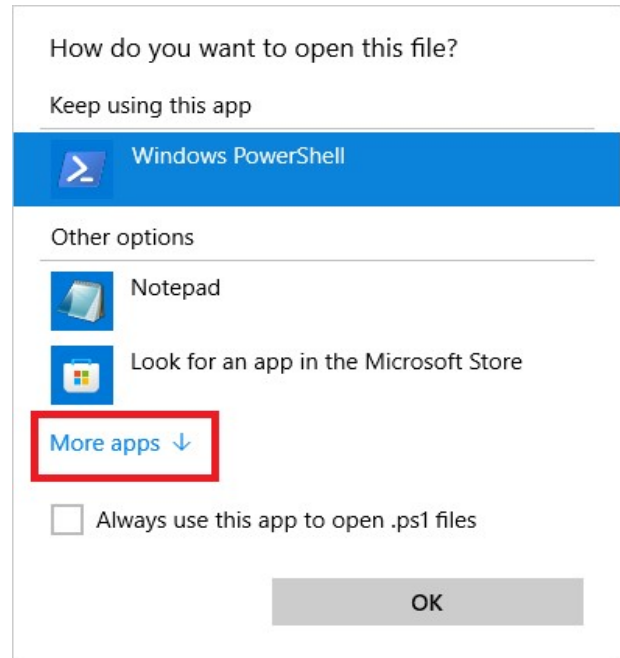
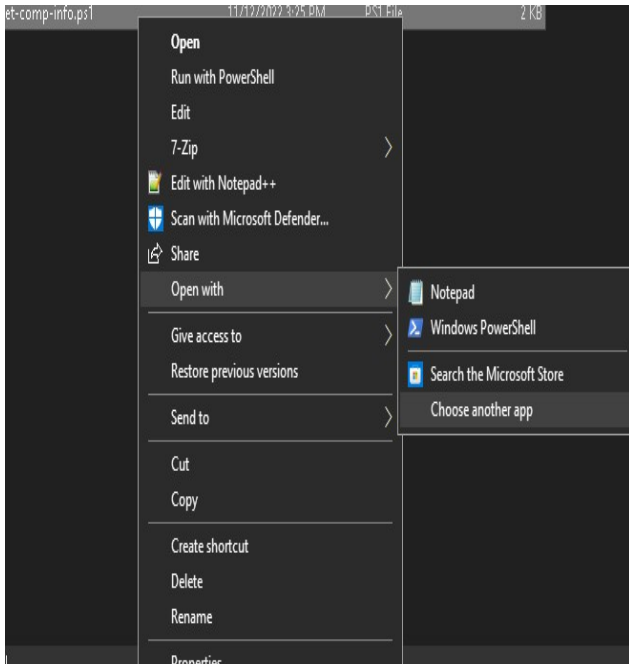
All shells have *built-ins*, or commands that are built into the shell. PS has a large number of built-ins and the ability to import third party modules, pieces of reusable code or code snippets, as well as your own. PS has four kinds of commands: scripts, functions and methods, cmdlets, and native commands. PS scripts end in .ps1 and are a list of commands, much the same as batch files and they work the same way. Functions are blocks of code written in PS, while methods are blocks of code written in other programming languages and called by PS. PS is based on .NET Framework. Cmdlets, pronounced 'command lets', are small blocks of code that perform specific things. Native commands are built-ins and work like cmdlets, but since they are built into the host, they are can be stand alone executables whereas the cmdlets are just instances of .NET classes^Φ.

Since we are talking about scripting, I will leave that as an introduction and let you explore further on your own by following the links below. The first thing you'll notice is you don't have to start with anything special

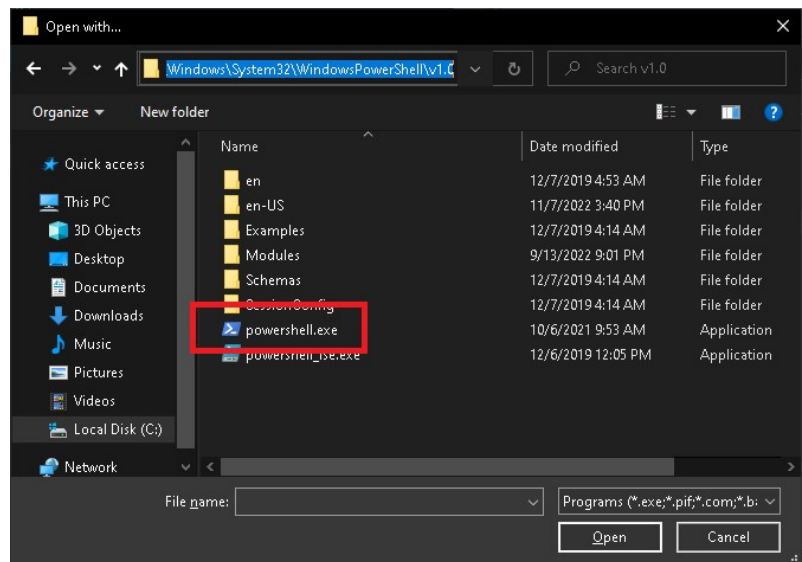
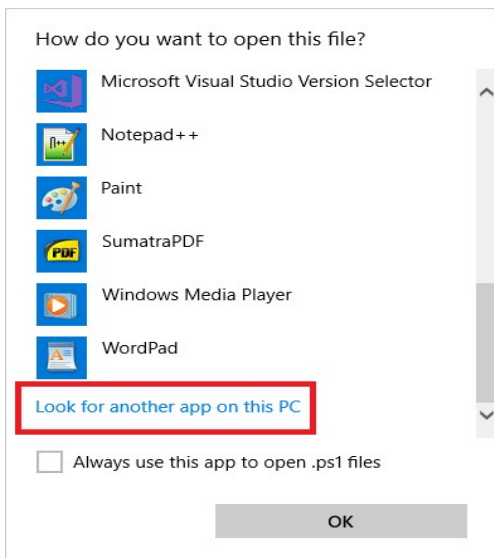
[∞] <https://learn.microsoft.com/en-us/training/paths/powershell/>

^Φ <https://learn.microsoft.com/en-us/powershell/scripting/learn/ps101/00-introduction?view=powershell-7.2>

such as the '@echo off' of a batch file, or the #! of a Linux shell script, just name it with a .ps1 file extension after changing the file type to read *All Files (*.*)* just like with a batch file. You may have to right click on it and select *Run with PowerShell* if it just opens up in notepad. You can set the default to open with PS by right clicking on the script and selecting *Open with*, then *Choose another app* on the box that appears. This will bring up another box that asks how you want to open this file.



Click on *More Apps*, and then scroll down to the bottom and select *Look for another app on this PC*. Then a window will pop up asking you to choose the file. Drill down into C drive, then Windows, then System32, then WindowsPowerShell, then v1.0, and select *PowerShell.exe*.



This will give your PS scripts the PS icon of the blue background with the white cursor and underscore and execute them in PS instead of notepad.

Open notepad and type the commands in, then save it as *filename.ps1* (whatever you want to name it) making sure to change the file type to all files, and that's it. You can also create menu driven scripts and asking for admin credentials is much easier thanks to the *get-credential* cmdlet. The way you call a cmdlet is by using a *verb-noun syntax*, such as *get-command*. This will print to the screen a list of all the available commands in PS separated by alias, function, and cmdlet. There are a lot of them. This will be our first script example.

Open a notepad and type *get-credential*, and on the next line type *get-command >> .\commands.txt*. Save it as *get.ps1* and change the file type to all files. When you click on it, it should ask you for a username and password, open a PS window for a few seconds, and when it closes there should be a file called *commands.txt* in the same directory as *get.ps1*. Notice how we accomplished in one command what took 19 lines of code in our batch file. Also notice that writing the output to a text file is the same as in a batch file. Using *>>* appends the text to the file, or adds the text to the end of it, while using *>* overwrites the text that is already there. This command did not require admin approval but was just an example of how easy it was to do in PS.

In order to create a menu driven script like we did with the batch file, we just write a function that creates the menu and then performs the actions, save it as a *.ps1* file, then click on it. I found this example on the internet at Tech Expert Tips where they have many useful scripts^ψ.

```
function DisplayMenu {
Clear-Host
Write-Host @"
+-----+
|           PowerShell Console - User Menu           |
+-----+
|                                                     |
|  1) ping                                           |
|  2) Display Message                               |
|  3) Exit                                           |
+-----+
"@
```

^ψ <https://techexpert.tips/powershell/powershell-creating-user-menu/>

```
$MENU = Read-Host "OPTION"

Switch ($MENU)

{
1 {

#OPTION1 - PING

$OPTION1 = Read-Host "Destination"

Test-Connection -ComputerName $OPTION1

Start-Sleep -Seconds 2

DisplayMenu

}

2 {

#OPTION2 - DISPLAY MESSAGE

$OPTION2 = Read-Host "Message"

cls

Write-Host "MESSAGE: $OPTION2"

Start-Sleep -Seconds 2

DisplayMenu

}

3 {

#OPTION3 - EXIT

Write-Host "Bye"

Break

}

default {

#DEFAULT OPTION

Write-Host "Option not available"

Start-Sleep -Seconds 2

DisplayMenu

}

}

}

DisplayMenu
```

I usually re-write these to make them original, but I don't know PS like the command prompt or bash so this is copied with a small change in format. As this is a technical support book and not a system admin book that is about it for PS instruction. PowerShell is a very powerful tool and I encourage you to use the links provided and learn as much about it as you can.

Software Installation

As with most other things related to computers, software installation is also not interchangeable between the operating systems. If you install the proper tools, you can compile your own software from source code in all operating systems, although the tools and techniques are slightly different. The core concept is the same in them all; software is a computer program designed to do something. It can be a game, internet browser, accounting software, or one of many other categories. It is either proprietary, meaning that the source code is private, or open source, meaning that the source code is available for inspection and modification, depending on the license restraints.

When you write software, it has a license and you choose what that license says others can do with it. US Government software cannot be licensed, it is automatically in the *public domain*, available to the public for free. You can also put your software in the public domain and nobody else can license it. There are a variety of licenses which range from trade secrets to public domain, and you need to make sure you are following the terms of the license for the software you use so you don't get in trouble. I have seen companies abuse the licenses that prevented it from being used commercially, as well as pirated software, which is illegally obtained, usually copied.

For example, the PowerShell script I used for the menu was copied from a website on the internet, changed a little bit, and a link was provided showing where I got it. When I finish the book, I plan on contacting someone at all of the websites I used for references for permission to link to it. This is always the best idea, but if you don't contact them and ask, at the very least link to it, leave any form of ownership information intact, and abide by the license. This book and the entire course are under the Creative Commons non-commercial with attribution license. It is pretty self-explanatory; anyone can use it, modify it, or whatever as long as they don't use it commercially and they have to give me credit for it. Information should be free, but that doesn't preclude one from getting paid for the work they do.

Windows

Recall that Windows installs software into the Program Files folder for 64 bit software, and 32 bit goes into Program Files (x86). Sometimes the program will need access to restricted places, and in this case, UAC should ask for elevation during installation, unless the programmer didn't write code to catch that like we did with our elevated batch file. When

installing software, the most important part is making sure you trust it. Malicious software, aka malware, is software that does things that are malicious, such as using your computer to mine bitcoin, load a backdoor into your system, or log the keystrokes you type and send them to a criminal. We will be going into more detail with malware later in this section.

Windows uses several file types to install software; .msi, .exe, .vbs, .msu, .bat, .ps1, and a few others which are less common and used by the installer. The .msi is a Microsoft silent installer package, .exe is an executable file, .vbs is a Visual Basic script, .msu is a Microsoft standalone update package, and you should know what a .bat and a .ps1 are, but in case you don't they are batch files and PowerShell scripts. By default, Windows does not allow software installation in Safe Mode, which loads only the necessary system drivers to start Windows, mainly used for troubleshooting and malware removal. We will go into more detail on safe mode in the troubleshooting section. You can add a registry key to allow software installation in safe mode, simply open an elevated command prompt and type

```
[HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SafeBoot\Minimal\MSIServer] @="Service" and hit Enter.
```

Programs can either be completely self-contained or they can use the libraries of code already installed on the system. These libraries are called *dynamic linked libraries* or .dll files. This allows programs to share code rather than having to install code for each of them individually. If the program needs code that isn't already installed, it can put its own .dll file in the System32 folder. This is an example of a program needing admin privileges since it is accessing the restricted area of Windows. By using shared libraries space is saved since each program shares access to commonly used functions; there is no need to duplicate that code for each program. It also helps with security since it reduces the amount of code used and centralizes it allowing for easier control and inspection.

The down side of that is that it introduces a *single point of failure*. In order to help mitigate this, .dll files are digitally signed, meaning they have a way to prove they are legitimate using a *cryptographic hash*. I have to take a small detour here to explain some things before I continue.

Cryptography deals with hiding or obscuring data and verifying the integrity of data. Using a code to write a letter that nobody can read except the person with the key is an example of cryptography. Another example is a hash, a series of calculations done on some data that produce a fixed length string as output. In order to be a secure hash, it has to be irreversible and can't have the same output from two different inputs,

known as *hash collisions*. We use file hashes regularly to ensure data integrity after file transfers. Since changing even one bit of the data being hashed changes the output drastically, after copying a file compare the hash to the original.

I use MD5 for simple file transfer checks, but it is broken and shouldn't be used for things like forensics or anything needing a legal *chain of custody*, or paper trail that will stand up in court. For this use the SHA2 family; six file hashes that give an output of 224, 256, 384, or 512 bits, while MD5 has an output of 128 bits. The reason I said MD5 (message digest) is broken is that there are known hash collisions. You can craft specially designed data that has the same hash as another file. The calculations are not as intensive as SHA256 (secure hash algorithm) so if all you need is to make sure the file you transferred didn't get corrupted, use MD5. If you are taking a forensic image of a hard drive for an investigation, use SHA512 and a hardware write block.

Windows has a built in hashing function called *certutil*. You use that from the command prompt and the syntax is: `certutil -hashfile <path-to-file>` (you can also drag a file to the command prompt window and drop it) `md5`. Certutil also has lots of other features~ but this is the one I use most. You can also check the hash with PowerShell, but from what I am learning, you need a 3d party module installed or use the .NET framework class with complicated code[^]. I don't like to use 3d party tools if Windows has the same functionality built in, so I rarely use PowerShell for hashing since the command prompt is so much easier. You can do some other things with PowerShell that you can't with command prompt, but again, the addition of 3d party tools or complicated code makes that a solution for scripting, not a one-time check of a hash after transferring a file to check for corruption.

Encryption is the hiding or obscuring part of cryptography. You don't need to know the difference between symmetric and asymmetric, or Blowfish, ROT 13, Diffie-Hellman, RSA, Elliptic curve, or any of that; just like PowerShell, entire books can be written about encryption. We were talking about digital signatures though. They rely on PKI, or public key infrastructure, a type of asymmetric encryption that uses public and private keys to create signatures, used most often to set up symmetric encryption sessions for the first time. You have two keys, the private and the public. When you create a message, you take the message and the public key and encrypt it. Then anyone who has the private key can

[~] <https://learn.microsoft.com/en-us/windows-server/administration/windows-commands/certutil>

[^] <https://devblogs.microsoft.com/scripting/learn-the-easy-way-to-use-powershell-to-get-file-hashes/>

decrypt it. This verifies identity and allows a symmetric session to be setup, using fewer resources and computing power.

That is how a digital signature in software works, except it uses a *certificate authority*. If you were a software company, you would send a request to the certificate authority and get a digital signature you would ship with the code for security. If anyone were to wonder if the software had malware hidden inside, they just verify it with the signature. In PowerShell, use the *Get-AuthenticodeSignature* cmdlet to verify it. At the very least, you should compare the MD5 hash to the one where you downloaded it from to make sure it isn't corrupted before installing.

Now, back to software installation. Now that you know how to verify the integrity of the downloaded software, both for corruption on the download and that it is the package that the company released, time to install it. When I said the software gets installed in Program Files that was simplified based on what you see when it's installed. Recall that Windows uses the registry in a hive, also known as a database. This is where the magic happens. It goes something like this:

1. Click on the icon of the downloaded software.
2. If it needs access to restricted areas of the system, it will ask for UAC elevation. Sometimes it happens at the beginning, sometimes after the process starts.
3. The zipped (compressed) files are unzipped into a temp folder.
4. The installation checks to see if there are any needed libraries (dll files) already installed, and if not, it installs them.
5. At almost the same time, it checks for any other dependencies, and if needed, they are installed.
6. While installing, the program is making the needed entries in the registry, and that is what makes the additions to the program files folders, desktop shortcuts, and making an uninstall script. This is how Windows keeps track of what was added so it can be taken out when you uninstall it.
7. Cleanup. Deleting the temp files, making sure everything is installed to the correct location, and configurations are updated.
8. At some point in the process, Windows writes the log entries to document the process with information entries in the event viewer. Reviewing the logs will let you know where to look if you have issues during installation.

This is just a simplification of the actual process. I encourage you to explore further if you are so inclined, but this will get you a level of understanding sufficient to troubleshoot the process. One thing that can cause problems is something you would never think about otherwise: cosmic rays. They come from outer space and can cause something known as *bit flips*. This is when a bit, 1 or 0, gets reversed or flipped. Most of the areas in your computer can handle a few of these with no issues, but get one in the wrong place flipped and you have corruption of data. This is one area of computer repair that is hard to identify and hard to fix manually. We will discuss it further in the troubleshooting section. Usually the fix is to uninstall then reinstall the software. If you have problems that are unexplained otherwise, it could be cosmic rays causing trouble.

Error Logs

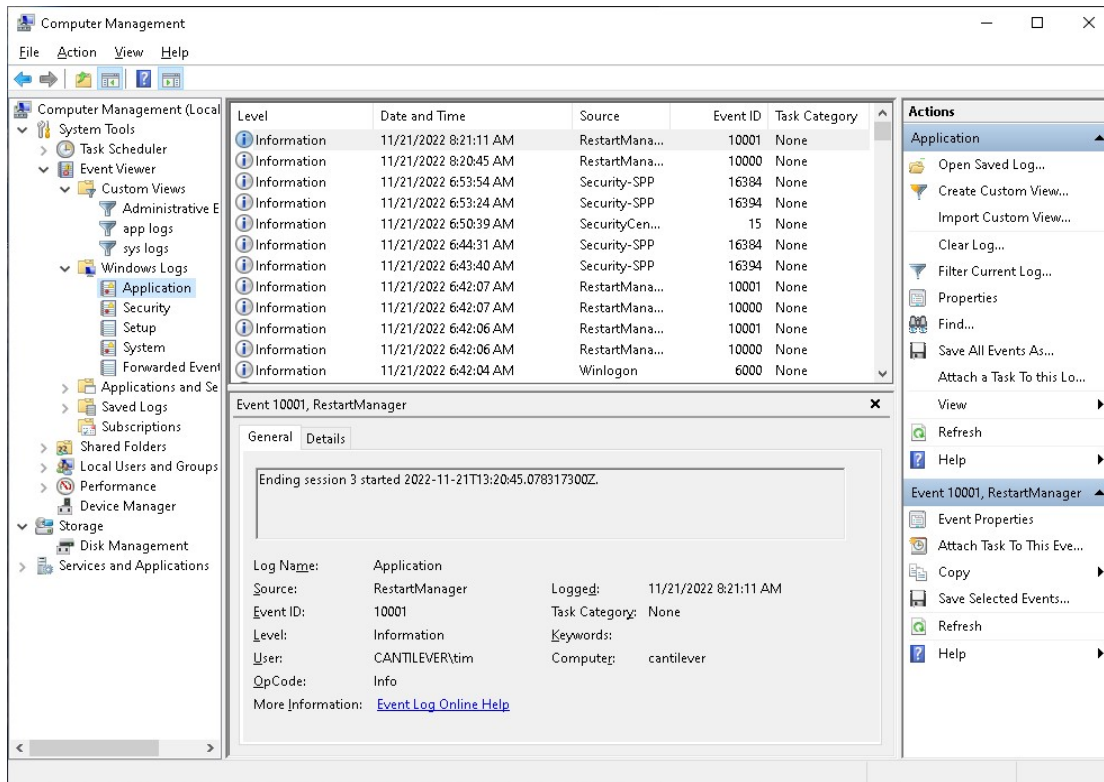
It can seem a bit overwhelming when you think about troubleshooting a computer problem. It makes it just a bit easier knowing that the computer will tell you what's wrong with it, you just have to know how to listen. All operating systems have logs, error logs, audit logs, access logs, and transaction logs among others. We are interested mainly in error logs, but we need to know how to correlate the other types of logs when troubleshooting. This allows us to put together a detailed view of what happened in the event that the error code is not obvious. When troubleshooting, the error logs are often the first place to look for answers. Although they may tell you what is wrong, sometimes it takes a bit of digging to find out.

The log files are only good if they are written well. Sometimes software makers do not anticipate the type of error and have not written an exception for it. In this case, the software can either crash, throw a generic error without enough detail to show the cause, or freeze and can even lock up your system. This is the reason we need to be familiar with error logs and be able to use them to deduce problems that are not immediately obvious. The more familiar we are with the processes we are troubleshooting, the better chance we have of quickly finding a resolution for it. The hardest part sometimes is the same as in life in general, just letting go. I tend to get wrapped up in finding the resolution that sometimes I spend way too much time trying to figure out the problem when the resolution is just uninstall and reinstall the software.

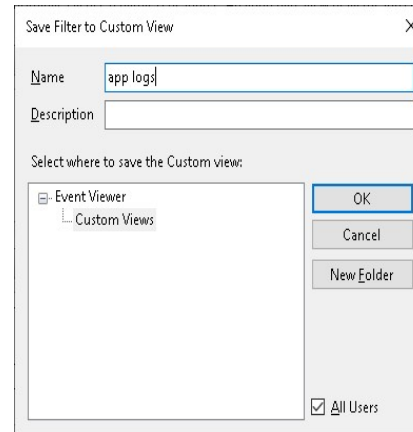
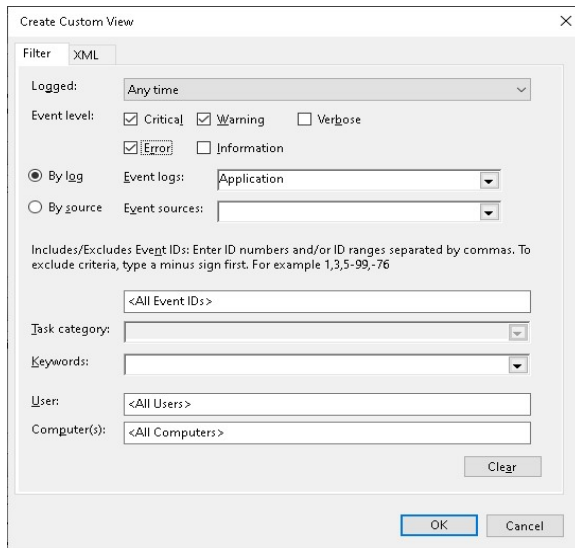
It comes down to the time thing again. If I am working at a call center on a remote computer I have the user walk me through what happens. This does several things; it gives me a chance to see if the user is making a mistake, it familiarizes me with the software in case I've never seen or used it before, and it gives me a timestamp to start searching the logs for an error. A reinstall will fix a lot of issues that you can't identify and save time on a call. Just make sure to always ask the user if there is anything they need to save before uninstalling anything. That is the number one rule before starting anything on a user's computer. Ask if they need to save any work before closing everything down and if they need to save anything off of the computer in case it gets erased. That will be the first step in the troubleshooting process and you should learn it well.

Windows

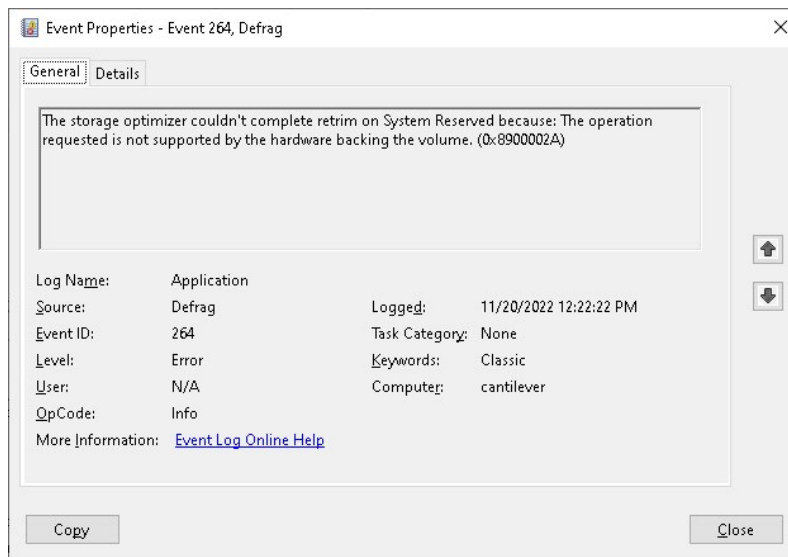
Windows error logs are viewed through the event viewer and have .evtx file extensions. They are stored in the %systemroot\System32\winevt\Logs folder. Recall the %systemroot is the root of the file system, or C:\Windows folder. Recall from page 31 that you can choose event viewer from computer management. The best way to find an answer is to have the user recreate the issue while you are watching, making sure to ask questions so you know exactly what is being attempted and the expected outcome, then take the error message that is displayed and typing that into the search engine exactly as shown.



Notice all the information events present in the image above. You can see in the pane on the left that Application is highlighted. If you right click on that and select *Create custom view*, this allows you to filter out the logs so you can see the bad stuff first. This saves time and gives you a starting point. When you find the relevant errors, then you can look them up on the internet in your favorite search engine, and hopefully find the solution. Most of the time you find a forum where folks are having the same problem and you find a few things you can try and see if they fix the problem. If not, then you go to the timestamp of the event and look for some information logs preceding it to see what was happening right before the error.



I like to name my custom views app logs and sys logs. I don't bother creating a custom view for security, setup, or forwarded events because I rarely need them. When I worked at a help desk, I always knew I had been on a computer doing work if I saw those custom views in the list.

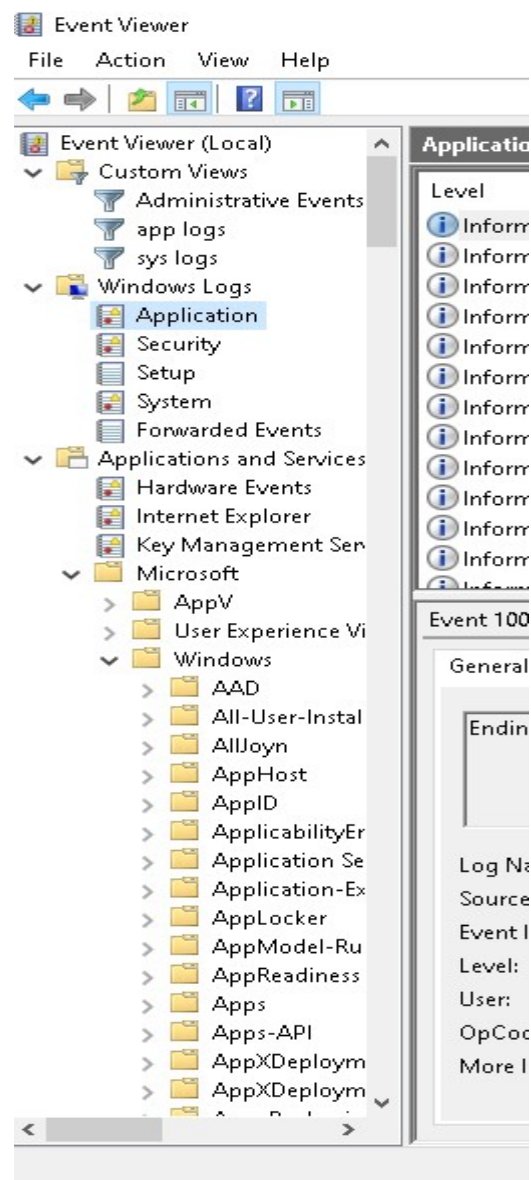
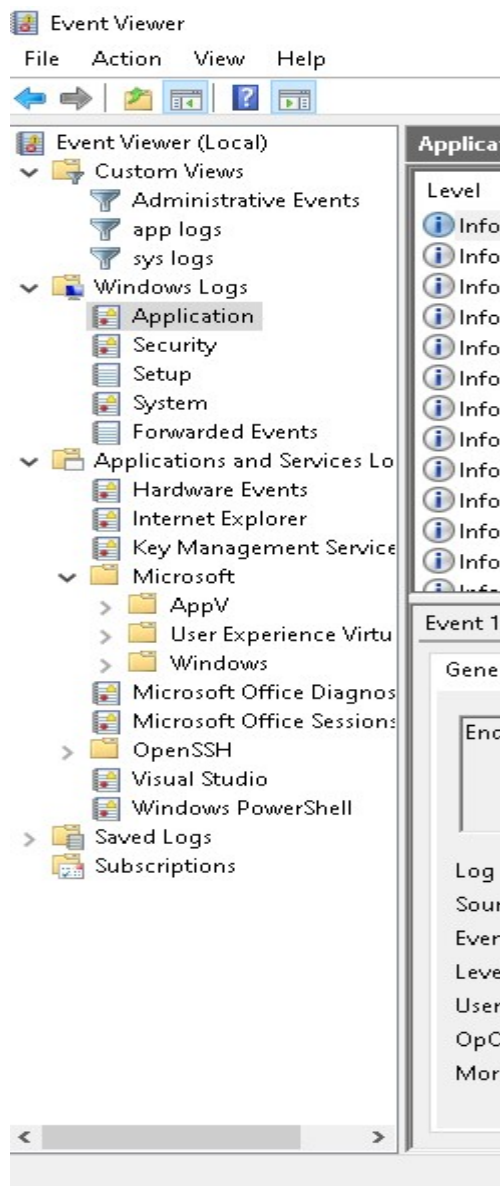


If you double click on an entry, it opens up the .evtx file and you can copy the entire message in the box and use that to search. Alternatively, you can use the link for online help, although that usually doesn't help a lot or search for things like "event 264 windows 10 <whatever the user was using or doing at the time>" and see what happens. Sometimes you have to get creative and use things like "Google dorks", which will be explained in detail later.

Notice in the *Actions* pane on the right side of the event viewer it gives you several options to interact with the logs. The filter and find options are how you search for things. It also allows you to clear the logs. I suggest that if you are working on a computer, research all of the errors and see if they need fixed or not, then fix the ones that need

it and clear the logs. I always like to save a copy to a work folder on the desktop or someplace on the user's home folder. This folder is used to download tools and programs to when working remotely on a computer and will be explained in detail further, but basically it keeps everything together and allows you to delete everything after completing the work.

Another section you want to pay attention to is the *Applications and Services* section in the left pane. This is where you want to go for events on Windows system components if you can't find a log entry anywhere else. Also, if you install the Microsoft program *Sysmon* from sysinternals suite, it is under the section shown in the second image when the Windows folder is expanded in its own folder located in alphabetical order. This is one of the best tools you can install for system monitoring and very useful in the event of a forensic investigation.



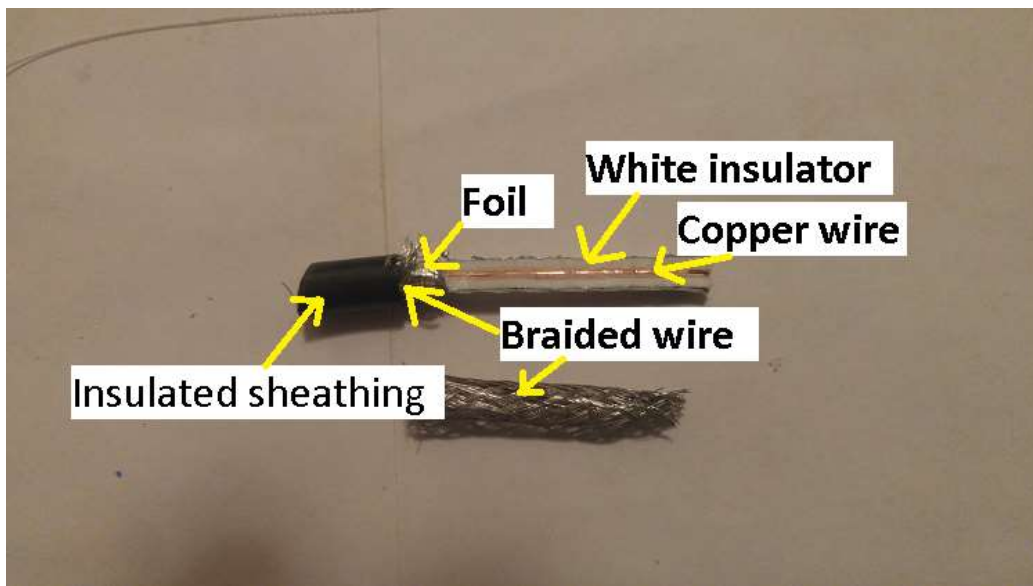
Networks

Material

The networks we are talking about in this book connect computers and other devices in order to exchange data. This data can be video feeds, the remote control of a power plant turbine, or a webpage with cat pictures. The network doesn't care; all it sees are a series of 1's and 0's, either in pulses of light from 860um to 1610um or pulses of electricity of varying voltages, usually limited to low voltage DC of 47V or less inside plant (the term for inside, whether it is a plant or a home is irrelevant). It consists of fiber optic cable or copper wire for wired networks and microwaves for satellite or radio transmission.

Wire

There are two types of wire used in networks; coaxial cable and twisted pair. The coax is the same type of coax used to carry cable TV signals and can carry up to 10 Gbps (gigabits per second) using MoCA (multimedia over coax alliance) technology, also known as *ethernet over coax*. It has an outer sheathing of insulation around a shield made of braided wire. This shield is wrapped around another insulator with a foil coat that covers a solid copper wire.



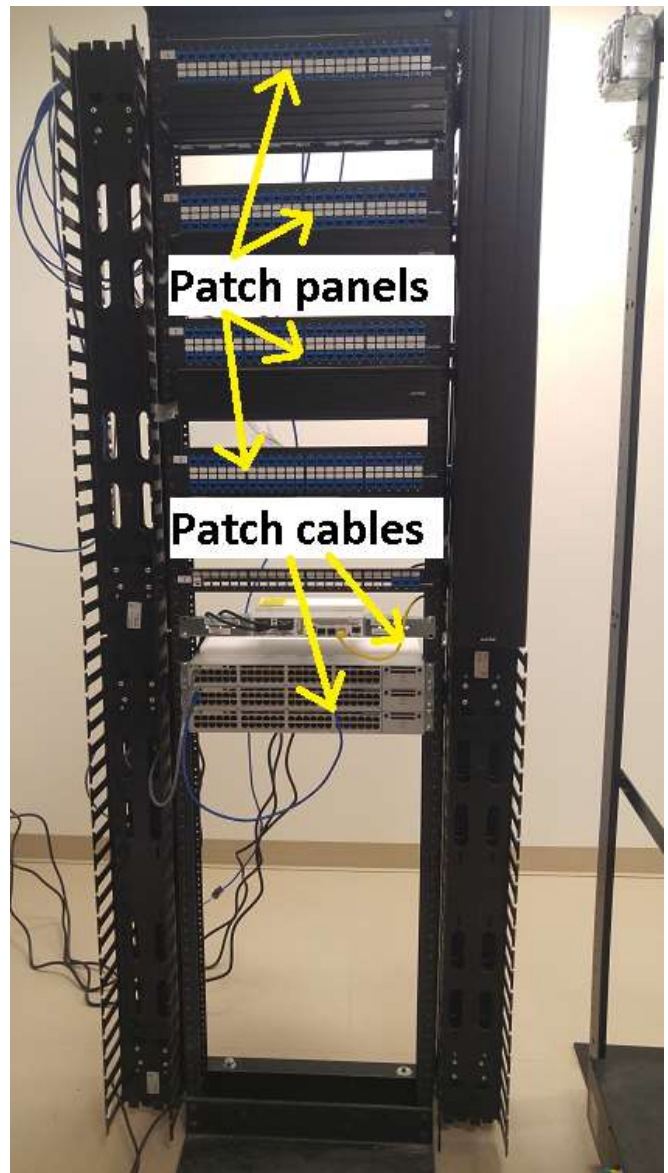
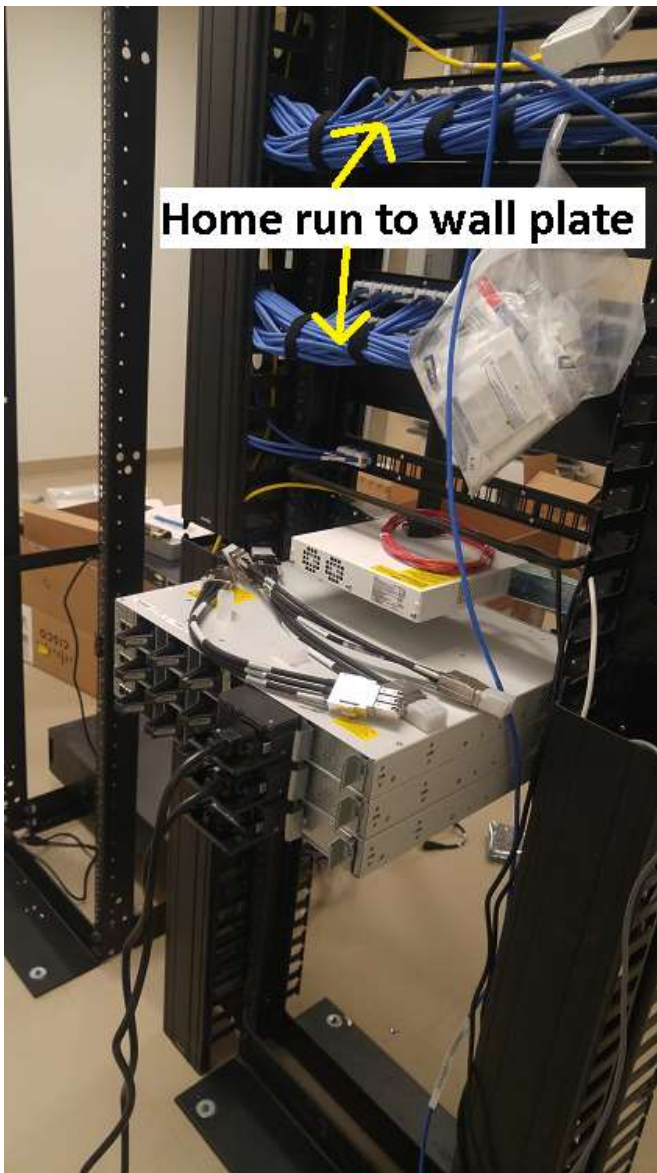
It has a very long range using repeaters but you won't get a very fast signal. It is limited to 100 Mbps (megabits per second) from signal degradation due to impedance. When you use coax, you need a *cable modem* to convert the signal to something you can use. This cable modem then feeds the signal to the *router* or it may have one built into the same

device. Then you hook up to the router either wirelessly or through a patch cable. If there are more devices than ports on the router, or if you are in a business environment, you will probably be using a *switch* to connect to. More on all of these in a bit, just know the general sequence for now. The coax is generally used *outside plant* and stops at the modem then *ethernet* takes over. The picture below shows the size difference in ethernet cable.



Twisted pair is also known as *ethernet cable*. It has 4 pairs of wires twisted together and is usually not shielded, so we call it *unshielded twisted pair*, or UTP. It has several standards, or categories, for the size of the wires and speed it can achieve. We call it Cat 5e, Cat 6, Cat 6A, Cat 7, and Cat 8. Cat 7 and 8 are used mainly in data centers for short runs. Cat 5e, 6, and 6A have a limit of 100m or 328ft, including patch cables on each end. Cat 7 and 8 are limited to 98ft. Cat 7 is not endorsed by IEEE, TIA, or EIA, but ISO^r accepts it. It is not backwards compatible with the connectors however. When you run cable, it needs to be one cable from the patch panel to the wall jack, known as a *home run*. The home run is limited to 90m or about 300ft to allow for a 5m patch cable on both ends. You can get away with a little bit over this limit since most patch cables aren't 5m on each end but not much more. It has to be a single cable with no breaks or splices due to signal degradation. When punching down the ends, for a cat 5e cable you can have a total of 1/2" of untwisted wire, for cat 6 you can have 1/4" untwisted, and for cat 6a it is limited to 1/8" untwisted and has to be certified.

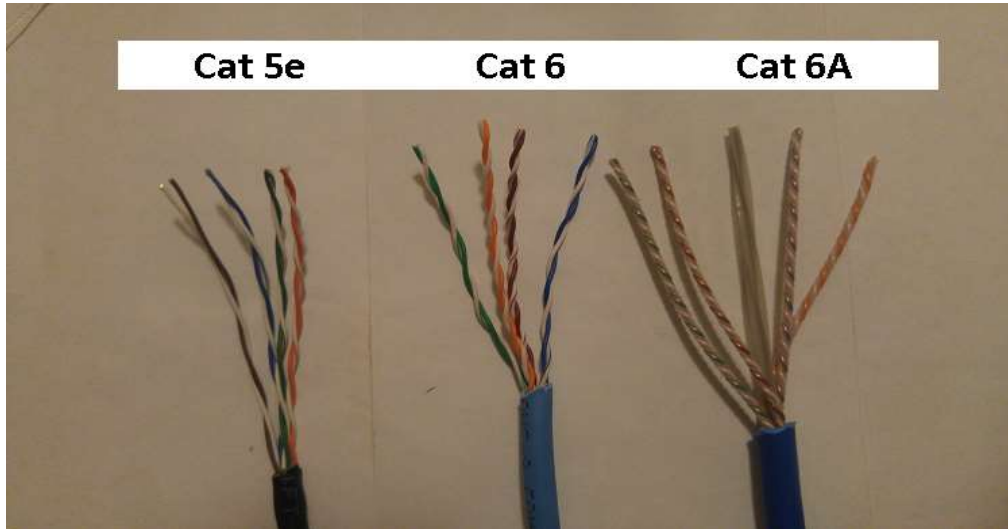
^r Institute of Electrical and Electronics Engineers, Telecommunications Industry Association, Electronic Industries Alliance, and International Organization for Standard. IEEE, TIA, and EIA keep the cables standardized while ISO let Cat 7 break it, so it is not very popular due to incompatibility of the connectors.



Now it's time to take a step back and go into some details about copper wire. Recall from the first section how electricity is made. Passing current through a conductor while it is spinning around a magnet causes electricity to flow through copper wires. The same thing happens in reverse also; passing electricity through a copper wire causes a magnetic field to appear around the wire. The wires are twisted together in pairs to help balance the magnetic field. If the wires were straight, they would be a large antenna and transmit the interference along with the signal. When twisted, any interference that affects one wire also affects the other wire in the pair.

Interference is caused when the magnetic field is picked up by one of the wires and not the other and is measured in the difference of the field between the two wires. When the signal is received, the noise is picked up by both wires and is cancelled out. Since it affects both wires at the same time, there is no difference in the amount of magnetic field on each

wire and no noise to measure. This noise is called *cross talk*, when the magnetic field from one conductor interferes with another. Twisting the pairs of wires helps prevent internal cross talk, but external cross talk is still an issue. Twisting the pairs tighter and using a larger diameter wire allows faster signals as shown in the picture below. Notice how tight the Cat 6A is wrapped and the plastic separator keeping the pairs apart.



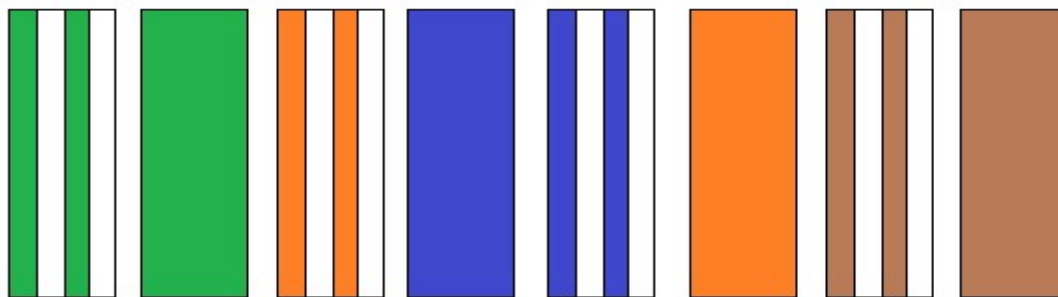
One way to prevent external cross talk is by wrapping a layer of foil or braided wire like used in coax around the pairs of twisted wires and grounding it on both ends. This gives the magnetic field a path of less resistance to follow. This is called *shielded twisted pair*, or STP, and is used in places where electrical interference is greater, like when cables are laid on top of lights or when ran on top of tiles in a dropped ceiling. They could have used some shielded cables in the picture below, half of which weren't even being used and were supposed to have been removed.



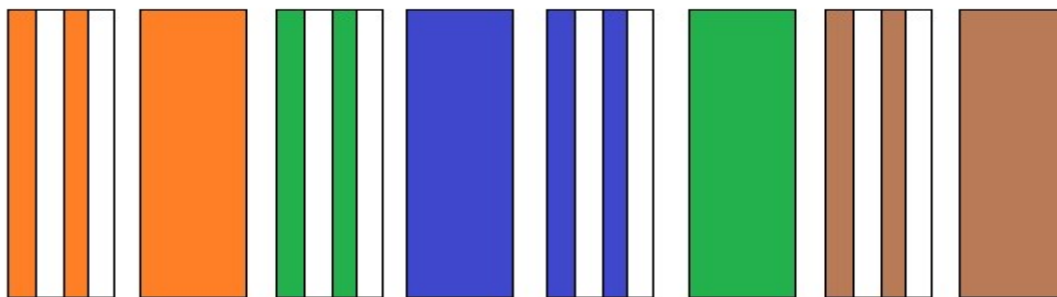
When investigating intermittent or hard to reproduce errors this should be one thing to look for as a possible cause. One time while troubleshooting an internet issue, I found that the modem was mounted on the wall with an electrical breaker box on the other side. Moving the modem resolved the issue.

Standards

You may have noticed that the colors of the wires in the pictures above were the same or nearly the same colors in all three cables. They use a color standard based on one that goes back to the old days of copper wires in telephone systems, and it carries on with fiber optics as a base for that standard. For ethernet, the standard is TIA/EIA 568 A and B. Either standard is acceptable since there is no performance difference, but always ask the client which is preferred or use the one that is there if not a new installation. I use the B standard unless the A is requested or used previously.



TIA-568A



TIA-568B

Notice how the blue and brown pairs stay the same. The orange and green pairs used to be receive and transmit, but now which pair does what is determined by the NIC, or network interface card. Please make sure you don't call it a NIC card; I had an instructor who instilled the hate of that redundant term into me so I graciously ask that you keep that tradition.

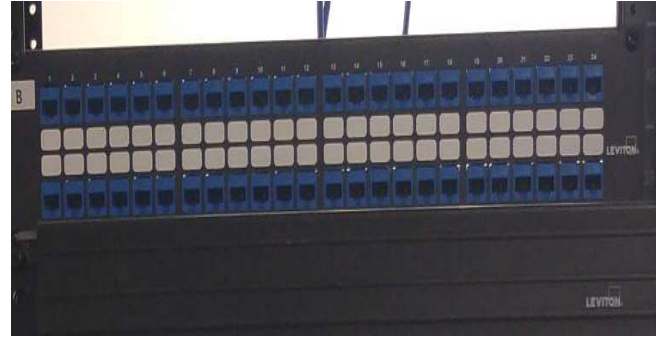
You can also use ethernet cable to provide power to things along with network access using PoE, or power over ethernet. You can carry 30W of DC using pairs 1 and 2, or 45W of DC in two channels using pairs 1 and 2 for 45W and pairs 3 and 4 for an additional 45W. Since there is a risk of power running along the wires, be careful when working with ethernet cables to avoid getting shocked. Since ethernet was not made to carry electricity, it has a higher loss, so not only is a 15% loss accounted for in the specification, manufacturers sometimes use more power than needed to compensate for this, around 60V.

You can also use BASE-T to talk about speed. 10 BASE-T is 10 Mbps baseband transmitted over twisted pair. The number shows how many Mbps the link is and if it were -F it would have been transmitted over fiber. 100 BASE-TX is known as fast ethernet, or FE. Gigabit ethernet is known as 1000 BASE-T, or GE. You'll see FE or GE on ports on routers and switches for the input.

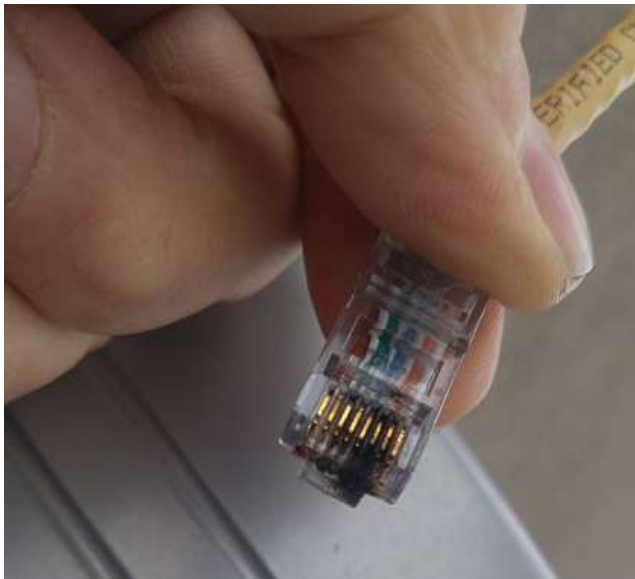
We talked a little bit about cabling and the 90m of the home run. When the cable gets to the patch panel, it has to be *punched down* by having the wires pushed into small grooves with sharp connectors that puncture the wires and are connected to pins on the other side where the patch cable goes. The other end of the run is *terminated* into a *module* that a patch cable plugs into. Both of these use a punch down tool while a crimper is used to attach the connectors. Notice how much wire is untwisted in this picture:



Patch cables use *8P8C modular connectors* which are almost exclusively referred to incorrectly as RJ-45 connectors. This most likely comes from the telecom industry as well since they call their connectors RJ-11 for the four pin type and RJ-12 for the six pin type. RJ stands for *registered jack*. In order to avoid confusion, I will continue to refer to them as RJ-45, since I did not learn that myself until writing this book.



The picture of the rack above shows an interesting feature, *modularity*. When something is *modular*, that means it is like a module or small block of something. The modules are interchangeable, and in this case, they use the modules as a patch panel. Instead of a 110 block, the cables are punched down into modules and the modules are installed into the patch panel. This saves time and money by using the same parts everywhere and allowing the cables to be installed faster and replaced easier. If you measure and mark carefully when installing, you can take the work to a field desk and sit down while punching down the modules and save the strain on your back. You'll thank yourself when you get my age if you do little things like that now.



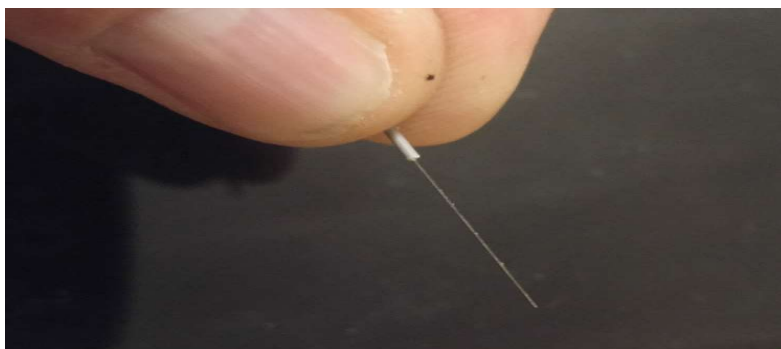
Notice two things about the above pictures. First, the lack of a drip loop has caused the water to short out the pins carrying the power to the security camera, which is why I was called out to repair it. Second, there is more than a half inch of untwisted wire on this connector. Obviously, (to me but maybe not so obviously to you) the short in the connector caused the camera to fail, but not so obviously, the untwisted wires could have been causing a signal degradation that was not reported

in my ticket since it was irrelevant at the time of the outage. As I wrote in my blog[©], when I arrived onsite I immediately went outside to the camera and found the problem since the root cause was improper installation, primarily lack of a drip loop for the cable, but secondary was lack of weather gasket. The drip loop would have stopped the water from entering the RJ-45 connector, but so would the weather gasket.

This is also a lesson for pay when working for yourself. I get a two hour minimum and travel since I had to drive over an hour just to look up and diagnose the issue in five minutes and repair it in 10. I was in the truck on the way home 20 minutes after arriving onsite with a successful repair. Had it not been for that two hour minimum, I would have been out half a day for 20 minutes of pay. Those lessons are usually learned quickly. We will address this and other real world scenarios during the troubleshooting portion of the book. I have a lot of them and had to take pictures of most of them. It's also a lesson in processes. Always start with the obvious, especially when troubleshooting intermittent issues. Reseat cables, check power, reboot, etc. One thing everyone does at one time or another is known as *over teching*, or over thinking the problem. Step back and get an overview first, then use your process, unless something jumps out at you, like the lack of a drip loop.

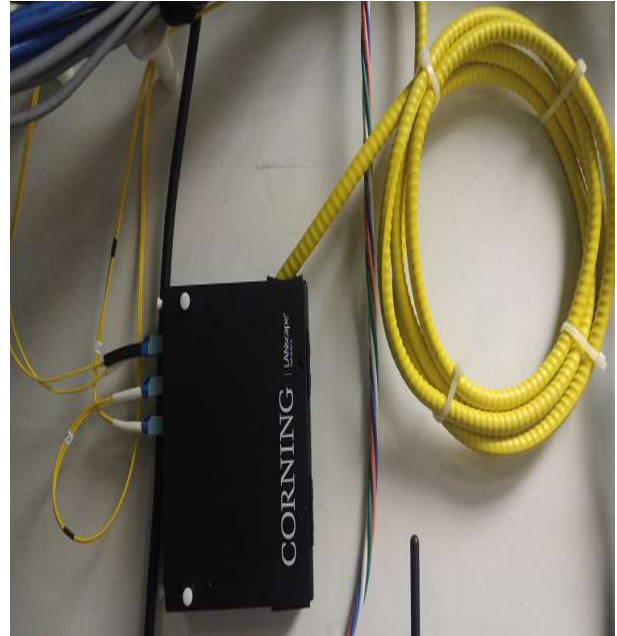
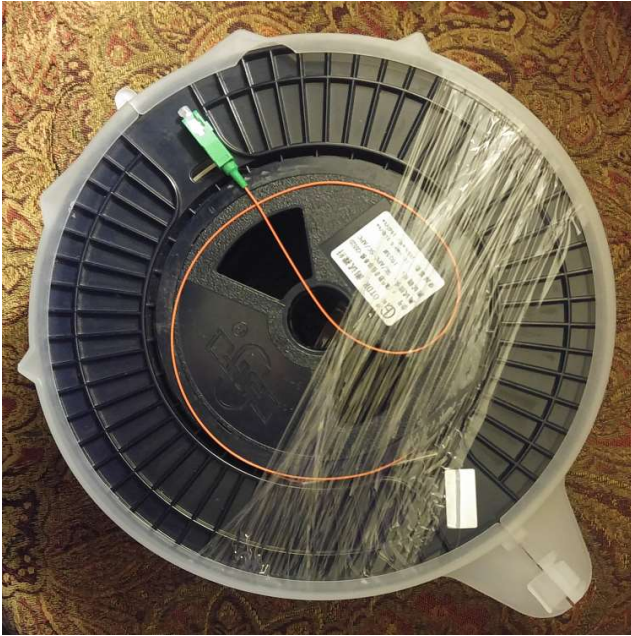
Fiber Optics

Fiber, *fiber optic cable*, is similar to copper wire in that it is in layer 1 on the OSI model (more on that in a bit), meaning that it is the physical medium that carries the signal, but that's about as far as it goes. It uses light instead of electricity to carry the signal and is made of extremely small strands of glass. It comes in single mode, SM, or multi mode, MM. Multi mode fiber comes in two sizes, 62.5um or 50um (micrometers). As a size reference, the human hair averages 40um to 50um. Single mode fiber is smaller than that, 9um or about one-fifth the size of a human hair! Notice the small specks of coating on the fiber. These need cleaned off before splicing. The glass is very strong, six times the strength of steel. But be careful with it, it hurts when you get stuck.



© <https://kitswv.com/thoughts220126>

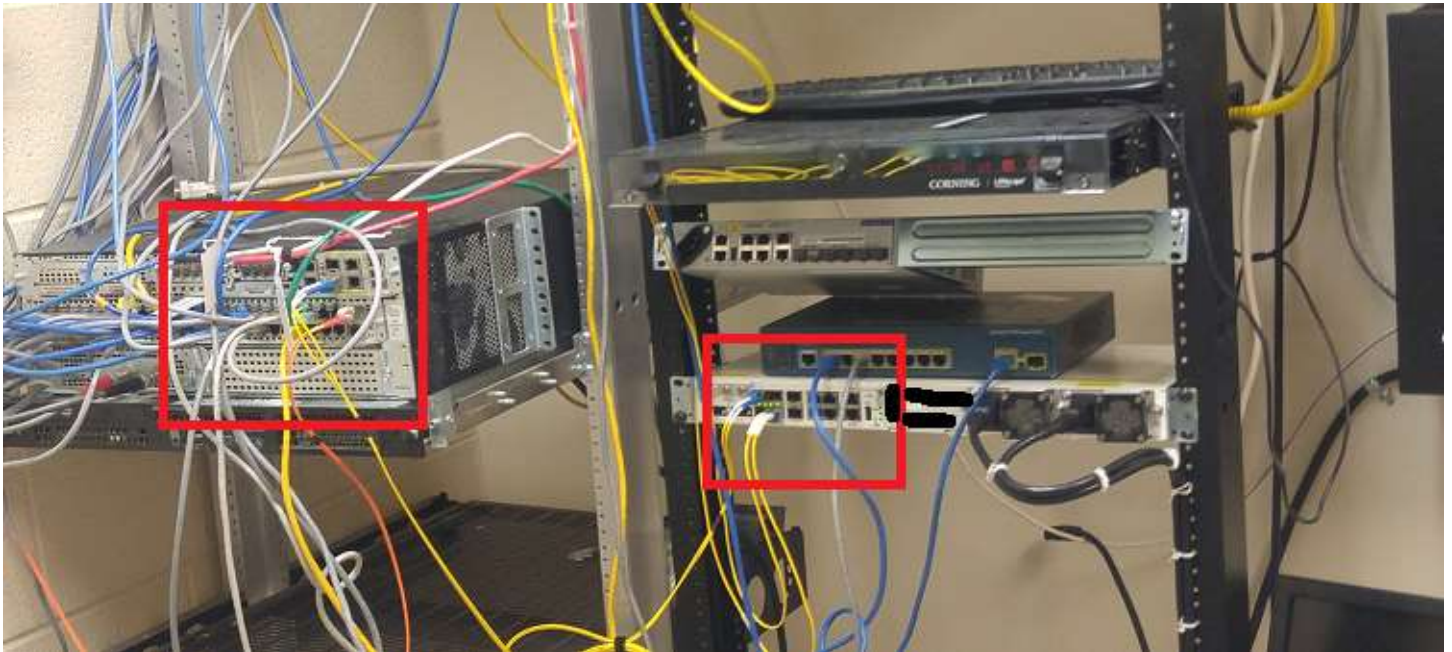
It consists of a core the light travels down wrapped in a cladding of glass with a different *refraction index* which keeps the light reflected inside and traveling down the core. Both SM and MM are wrapped in a 125um cladding and come in a variety of packages from single strand wrapped spools for test equipment to armored multi-strand cables wrapped in stainless steel with a plastic coating suitable for direct burial.



MM can carry several different *wavelengths* of light at the same time, but this limits the distances covered. SM only uses one mode and the distance is nearly unlimited. *Dense wavelength division multiplexing*, or DWDM, uses SM to send different wavelengths at the same time on SM fiber. This increases the bandwidth and allows one to transmit and receive on the same fiber. It is often used in *passive optical networks*, PON, in *fiber to the home*, FttH. FttX is used when x means office or something other than home.

Splicing fiber can be done two ways, mechanical splicing or fusion splicing. They both have advantages and disadvantages, and fusion splicing is preferred as it is better. Mechanical splices are cheaper since they don't require expensive equipment but aren't as good. They involve putting the fiber into a mechanical connector of some sort using an index matching gel to help the light move through the splice without as much loss or reflection as if there were nothing but air. This helps, but you always have reflection at the splice since the two fibers aren't touching. Fusion splicing actually fuses the fibers together and takes away the reflection. It shows up as a small loss, generally less than 0.1dB, *decibels* - the unit used to measure signal strength, but since there is no endface of another fiber with a gap, there is no reflection.

Fiber requires different equipment than copper, although they do make things that convert the light to electricity and vice versa called SFP's for small form-factor pluggable. These allow fiber signals to be converted for use in routers and switches. Notice how this is different from ethernet cables that just plug directly into the ports.



In general, copper requires a modem while fiber requires an ONT, or optical network terminal. Both do the same thing just with different types of signals. All of that will be covered in more detail coming up. I encourage further exploration of fiber optics and a great place to start is the Fiber Optic Association[√] where they have a lot of great resources for learning about fiber optics.

Topology

Topology refers to the logical layout of the network. This may or may not be different to the physical layout. For example, in a *token ring* network, the devices may not be laid out in a ring, but the cable makes a ring either in the walls or throughout the building in some way. This only means that it is a loop, not that it follows a circular or ring shape. There was a time when this was more important than today, but now we mainly use *star* networks where every device is connected to a central hub (we use switches now instead of hubs, but more on that later). Hybrid networks are a mixture of types. Star networks are most often seen in *enterprise* environments, or large business settings and schools. They usually have a central location such as an I T closet where the *demarc*

[√] <https://www.thefoa.org/>

(short for *demarcation* - the point where a utility such as gas, electricity, or phones enter the building) is located. Sometimes they have a demarc extension which is running a cable, either fiber or copper, from the demarc to the I T closet or network room.

In ring topology like the token ring mentioned above, data is sent around the ring in one direction, so traffic may not always follow the shortest path and a single point of failure is introduced. Of course, even in a star network a single point of failure is the switch; although they usually have more than one switch, since most of them are either 24 or 48 ports, meaning that unless there are fewer devices than that, more than one is needed. But that still leaves the router as a single point as well as the modem or ONT unless you have a backup on a different device such as a Cradlepoint or something similar. Most VoIP, voice over IP, phones these days are *daisy chained*[¶] to the computer so that users can use one network drop instead of requiring two. The phone acts as a pass through for internet traffic while also sending and receiving its own traffic. Just to be clear, this alone does not make it a hybrid network.

Another type of network is a *mesh* network. It is a type of *peer to peer* network where devices communicate directly with one another. This can be convenient when done correctly, but can also become a security nightmare.

Networks can also be described by the area they serve, either local or wide, and are designated WAN or LAN for wide area network and local area network respectively. You typically see a LAN inside a building or maybe a few buildings on a campus and a WAN across several cities. I have been installing SD-WAN systems in businesses in the area which tie in many different LANs using *virtualization* with software, hence the name. This is so that MSPs, *managed service providers*, can manage the networks of many different companies more efficiently while still segregating the networks. We'll be getting familiar with WANs and LANs later, and there are other designations - WLAN, PAN, SAN, MAN, EPN, and VPN, but they are abstractions that you don't need to worry about as a level one tech, other than familiarization, except for VPNs, *virtual private networks*.

In order to talk about VPNs, we need to go into two things first: virtualization and encryption. As mentioned earlier, encryption is all about keeping something hidden or private. There are several different protocols used but they all work in similar fashions by creating an encrypted tunnel to send information across the open internet without being seen. It could be a software VPN or a hardware device. Virtualization is using software to act like physical devices. Part of the class that this book accompanies is conducted using virtual machines.

[¶] daisy chaining means to connect one device directly to another instead of to a network drop.

In this case the virtual part refers to the connection. It is entirely created by the software or hardware device and isn't actually a private connection going from your computer to your work place or whatever website you're visiting. You can think of it as taking all of the data transmitted over the wires, encrypting it, sending it along to be decrypted before being used, and then decrypting the data coming in and reading it. This is simplified but close to what happens.

Some companies use a VPN for their employees who are working from home so they can be on the company network. VPNs can also be used to show your traffic as originating from the exit point of the VPN instead of your actual location. They are legitimately used by activists or others fearing their oppressive governments, or to access content blocked in their locations as well as for illegal purposes. The same can be said for a lot of the tools I'll be telling you about. One thing to remember, underneath all of it are the same ones and zeros, sometimes electricity, sometimes light.

Equipment

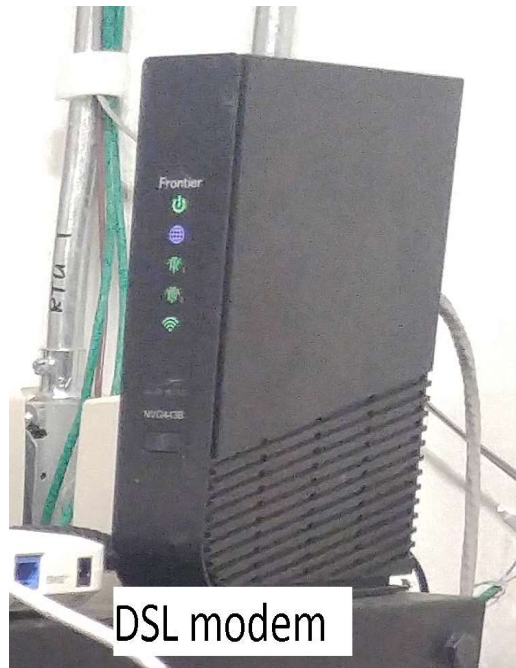
Before we go any farther I need to briefly mention the *OSI model*, or how we describe network traffic as it is handled throughout transit. Open systems interconnection, OSI, refers to the different layers of transport from the physical layer, layer 1, all the way to the application layer, layer 7. I learned to remember them using the phrase, "*Please Do Not Throw Sausage Pizza Away*" for Physical, Data, Network, Transport, Session, Presentation, and Application. We will go into more detail about what it is and how it works and applies to you later, but for now when I talk about a layer 2 device you know it works on the data layer and that a layer 1 device works on the physical layer.

I also want to give you a brief overview of the process. It started with four computers in 1969 when the first readable message was transmitted from UCLA to Stanford. In true internet fashion, when they tried it the first time only the first two letters came through. They had to fix a bug and transmitted the word "*Login*" an hour later. Now the number of devices connected to the internet is in the billions. It should be of no great surprise that with only a handful of computers at various labs and universities they didn't put much effort into security, a habit that plagues us to this day. Without becoming a history lesson, the important part to remember is that many of the protocols and standards we use today were developed almost 50 years ago, with the TCP/IP coming along in the 1980s. That stands for transmission control protocol/internet protocol. There have been many more since then but this is the bedrock which they

built the World Wide Web on in 1989, more specifically, IPv4, for version 4.

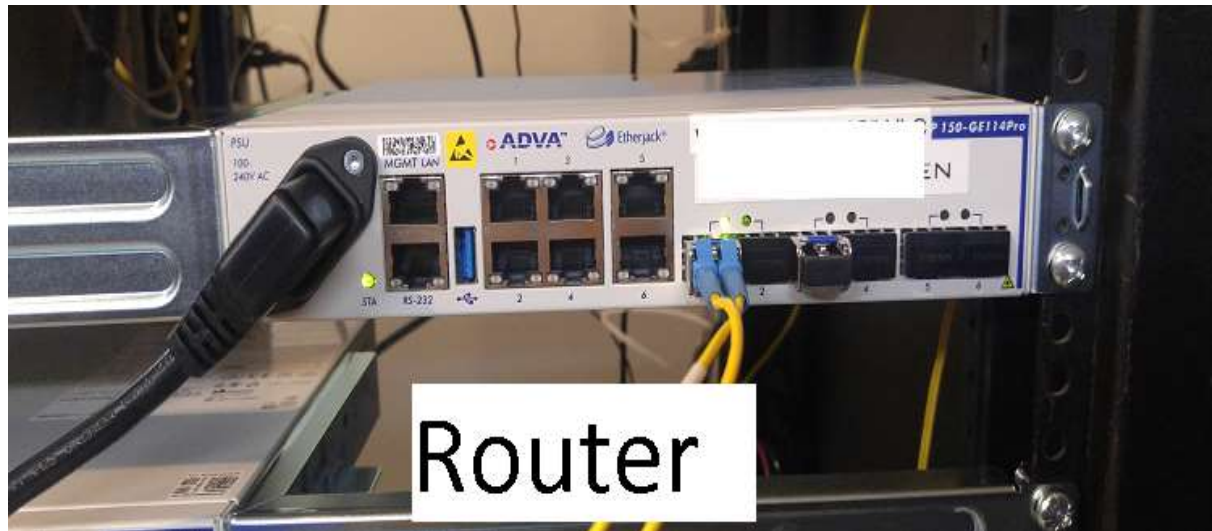
The first three were never implemented, so IPv4 is actually the first version of internet protocol, which uses four *octets*, or groups of 8 bits for 32 bits total. They are represented numerically in groups of digits from 0-255 separated by a "." and they look like this: 192.168.24.65 or 122.43.213.76. I'm going to go into more detail on this later, but know for now that numbers are easy for computers to read but hard for humans, so we came up with a naming convention called DNS, domain name system, to put names with the numbers. This is how yahoo.com translates to 98.137.11.164. This all works relatively well until you realize that when the internet started there were not that many computers.

They came up with NAT, network address translation, which provided a way to extend the public IP addresses into re-usable private IP addresses. Any IP address that starts with 192.168, 10, or 172.16-172.31 are private IP addresses, meaning that they are not routable on the internet. This is how we can use the finite number of public IP addresses for more devices than we have available addresses, in fact, we ran out of public IP addresses in early 2011. We now have IPv6 which has more than enough for now, but implementation has been at the pace of glaciers it seems. Again, we will go into more detail later, but this is enough to work out how network equipment works.



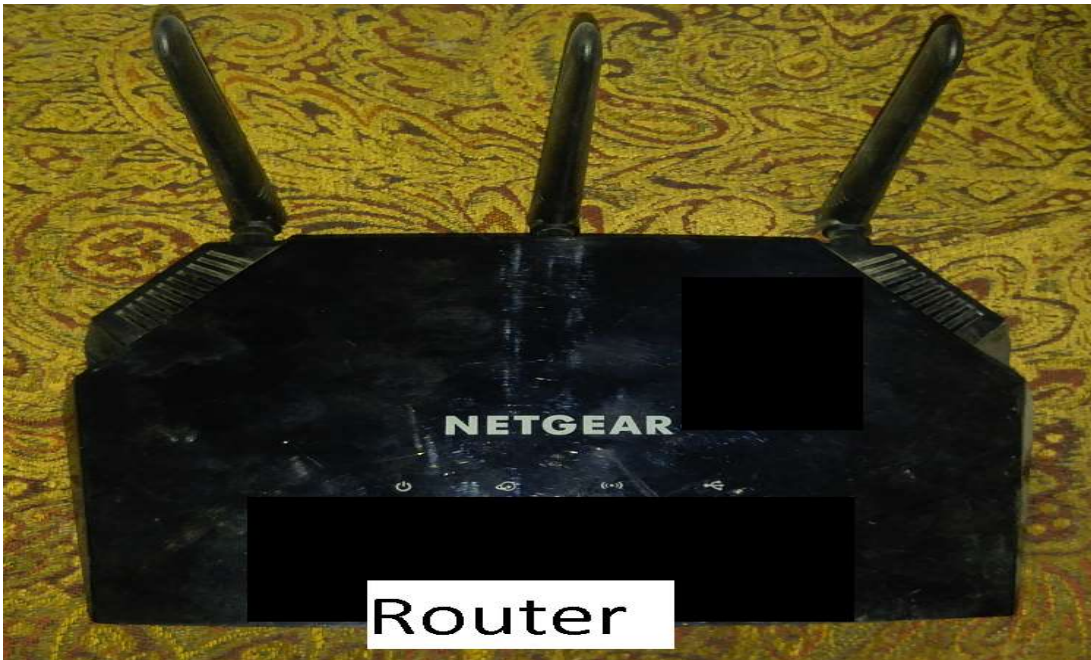
The first layer 1 device that comes to my mind is a *modem*, which stands for modulator-demodulator. In this case modulate means to change a signal in some way. It receives the data from the outside - through coax cable for cable modems, telephone wires for DSL, digital subscriber link, or

fiber optics for an ONT, optical network terminal, and demodulates it before sending it to the router. The opposite is true for outbound signals which get modulated before going out on the wire. This is a vast oversimplification of what actually happens, and I encourage you to learn more, as always. A great website I found that explains things pretty well is <https://www.howstuffworks.com/>. You don't have to use that site and I don't get anything for it except ensuring you have accurate information, but if you are reading about how a modem works and it doesn't mention the *MAC*, *QAM*, or *burst modulation*, then you should make sure to check your source for accuracy. I like to write out things as I recall them and then go check to make sure I didn't forget anything. I was a little bit surprised at how much of the information I am passing along to you I remembered from school, and also a little surprised at how hard it was to find details I needed to check, which is the main reason I am writing this.



Routers are where it gets interesting and the real work gets done in your network. They are a layer 2 device and if you have more than one device on a network, you need one to send the data to the correct device, or in other words to route the traffic, hence the name. Recall that we mentioned LAN and WAN as separate networks, and in the case of routers, we use them for internal and external networks based on the ports used and where the data is going. For home or small business routers, the WAN port is the external network, where the incoming signal from the modem comes from while the LAN ports are the internal network, where the signal goes to. A good way to remember it is that private IP addresses go to internal networks and public IP addresses go to external networks. You keep private things inside while public things go outside. For enterprise routers, the WAN port or ports may be labeled FE or GE for fast ethernet or gigabit internet, either by itself or with a series of numbers and slashes after it such as GE 0/0/0 or FE 1/1.

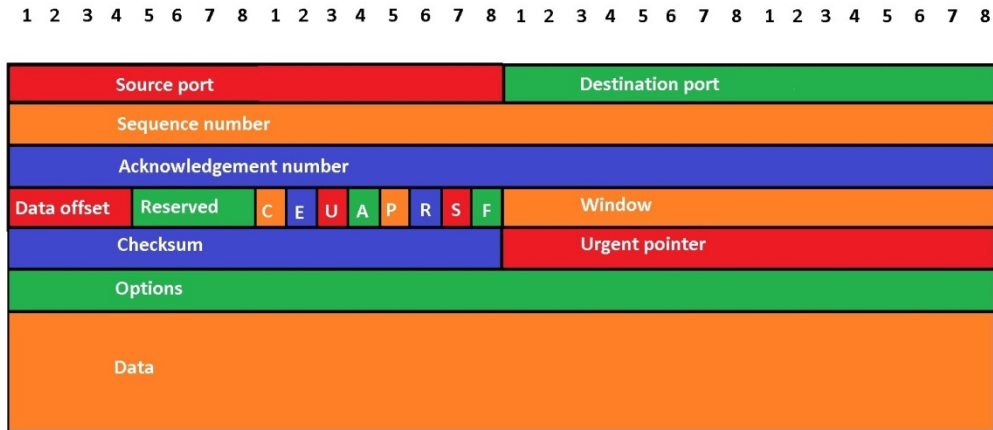
Routers use DHCP, dynamic host control protocol, to assign IP addresses from a pool of private IP addresses so that many devices can all share one public IP address.



Every IP address comes with a *lease*, which has a time limit and when it runs out, the lease ends and the IP address is then reassigned to the device or put back in the pool if no longer being used. *Static* IP addresses are typically set manually and mainly used by servers and printers since they are accessed frequently on the network. Different routers have different ways to do this, usually by setting a *reservation* for the address in the router or in Windows as mentioned previously.

In order to keep track of what device is assigned to which IP address, the router uses a *lookup table* and identifies the devices by their *MAC address*, also known as the physical address. It is written in hexadecimal format or hex for short. Hex is a base 16 number system intended to express binary numbers in a format humans can read more easily than a long list of ones and zeroes. It uses 0-9 and the letters A-F and I will explain it all in an upcoming section. For now just know that if you find a number that looks like 78:1F:DB:26:A2:6B, that's the MAC address and it's a Samsung device. The first three pairs of numbers are the manufacturer and the second three pairs are the serial number. If this were written in binary it would look more like 01111000 00011111 11011011 00100110 10100010 01101011. It's not very hard to see which one is easier for us to understand. You need to keep in mind that the MAC address is assigned to a network interface and a device can have more than one.

We will get into packets and frames in the protocol section coming up, but we need to touch on a little bit of it now to better understand how a router works. You know that data is transmitted as strings of ones and zeroes, but what we haven't discussed yet is how they are understood. *Packets* are just a group of bits, but the position of the bits acts like a sort of code book so the router understands what to do with it. Take the following illustration as an example:



TCP packet structure RFC 9293

The numbers across the top represent the number of bits.
 The sequence number, acknowledgement number, and options are all 32 bits.
 The source port, destination port, window, checksum and window are all 16 bits.
 The data offset and reserved sections are each 4 bits.
 The flags are all single bits and considered set if they are 1 and not set if they are 0.
 C is CWR - congestion window reduction
 E is ECE - ECN echo - explicit congestion notification
 U is urgent. When this bit is set, the address of the pointer is located in the urgent pointer field.
 A is ACK - acknowledged.
 R is RST - reset
 S is SYN - synchronize
 F is FIN - finish

This is what a TCP packet looks like, and we will be going into greater detail in the protocol section, but notice the top section where it says source port and destination port. There are 65,535 ports to choose from because this is the maximum number you can have with 16 bits - the size of each field. These are *logical* ports, not *physical* ports. Physical ports are what you plug a cable into. There are approximately 1000 common ports such as 23 for SSH, 80 for TCP, and 25 for SMTP⁹¹ but this is just a list of commonly used ports, not a rule. You can use 2280 for DNS if you like even though it is commonly found on port 53. Some use this tactic for added security, and it may stop some attackers, but generally security is like quantum physics in that there is no absolute certainty, only various levels of probability, or in other words, you can never be 100% secure,

⁹¹ I have included a list of common ports as an index.

only close. Given enough time and resources for the attacker, nothing can prevent a breach[∇].

This is all nice but you may have noticed it doesn't have any information besides the ports. That's because the router keeps track of that. Remember NAT? It's where the router uses one IP address, the public one, to share between multiple internal addresses, the private ones. The private IP addresses can't be used on the internet and the most common ones are 192.168.x.x, although 10.x.x.x is becoming more common and is what I prefer to use, if for no other reason than network segmentation. But I digress. The router uses its own wrapper to put on packets before sending them forward and strips them off before sending them to their destination. We will use Wireshark to look at these packets for real world examples in the protocol section.

Until now, I have been referring to routers in general, or ones you may encounter in a home or small business. Enterprise routers are more advanced and require detailed configuration, although they are moving toward GUI dashboards rather than traditional CLI interfaces. Cisco and Juniper were big names and still are, but Ciena, Adva, Adtran, and many others are starting to push out the more expensive enterprise routers for central management by an MSP, manage service provider, or company that monitors the network for security and reliability. They provide a more secure portal to the network than just using the customer equipment.

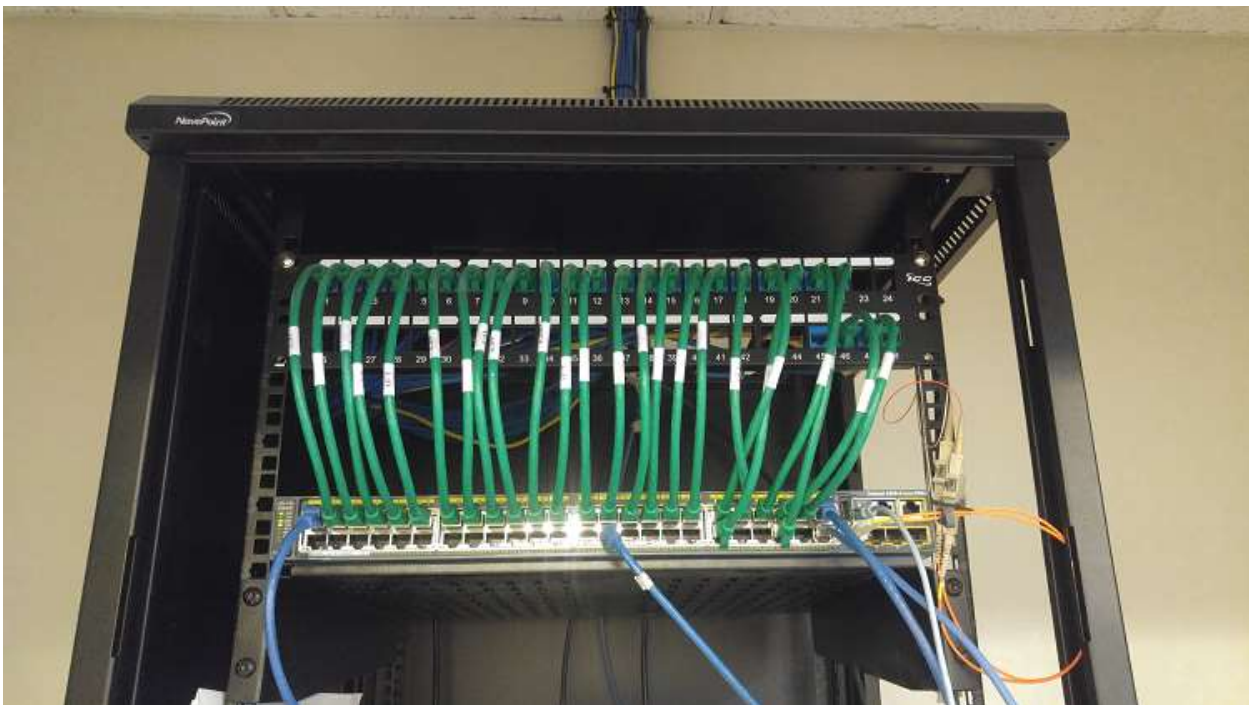
For network failure, you need to go onsite and use a *console cable* to hook up your laptop and use a terminal emulator such to gain access to the router. Some newer routers use a normal ethernet cable and connect to the console port, or even just a regular port, and access the dashboard through a web browser. Normally, I would say that a level one tech needs to know some basic commands for routers, and while that was definitely true 12 years ago when I went to school, it is starting to change and now has become less important due to the increase of dashboards mentioned earlier. I am going to include a list of common Cisco commands as an appendix because it is good information to have. As long as you have a laptop and a hotspot and know how to connect to a router with a console cable you can be a level one tech. I have done a lot of work that just required me to connect to the equipment and let them have remote access to the laptop and sit there and watch, maybe move a cable or flip a switch every once in a while.

[∇] As of the time of this writing. Something may come along in the future, but I seriously doubt it.

Switch

Switches are not so straight forward. They can be layer 3 (unmanaged switches) or layer 2 (managed switches). They used to be referred to as smart or dumb, but thinking like that is not good. I include it for reference only; some still call them that so you should be aware of the term. This is similar to how primary and secondary drives used to be called master and slave until we realized that using more descriptive language was more inclusive and just an all around good idea. Back to switches, layer 2 switches have limited routing functions, thus the "smart" descriptor. When you plug a device into a switch, traffic is sent to the router asking for an IP address (if you use DHCP), and the router enters the devices's information into the routing table and assigns an IP address for a period of time known as the *lease*.

Alternatively, *static IP addresses* use the same IP address for a device every time instead of grabbing one from a pool of available addresses. A switch allows *full duplex* communication, allowing traffic in both directions at the same time, by sending traffic to each device rather than just broadcasting everything. This is how hubs used to work, and then switches got cheaper so the cost to replace them was offset by the increased security, speed, and reliability.



This is an example of a managed switch. Notice how the pretty wiring from the patch panel to the switch distracts from the substandard workmanship leaving the connectors hanging in the air rather than installing a fiber patch panel. That \$75 addition during installation could save a \$500 service call in the near future; fiber is delicate and not cheap to work

with. Managed switches can do things like run VLANs (virtual LANs) to create network segmentation by creating virtual networks for different devices. This is one of the first areas you should look at when having network issues since VLANs can drop members through corruption just like computers or if someone plugs into the wrong port.

A good way to visualize what a VLAN does is to imagine having several different departments' computers plugged into the same switch, but each department's computer is isolated from computers in the other departments. As far as the computers are concerned, they may as well be in different networks and not the same switch. This is not to say it is perfect, and in fact, you can jump VLANs using different exploits, but it isn't easy. This is one reason why SCADA systems[€] are required to be kept on separate physical equipment at natural gas compressor stations.

ARP, address resolution protocol, is how switches talk to the router and keep track of the connected devices. When we look at the Wireshark trace, notice the ARP messages. They will say things like, "who has 10.10.125.122?" in order to keep track of devices. This is known as polling and is done at regular intervals.

Access Point

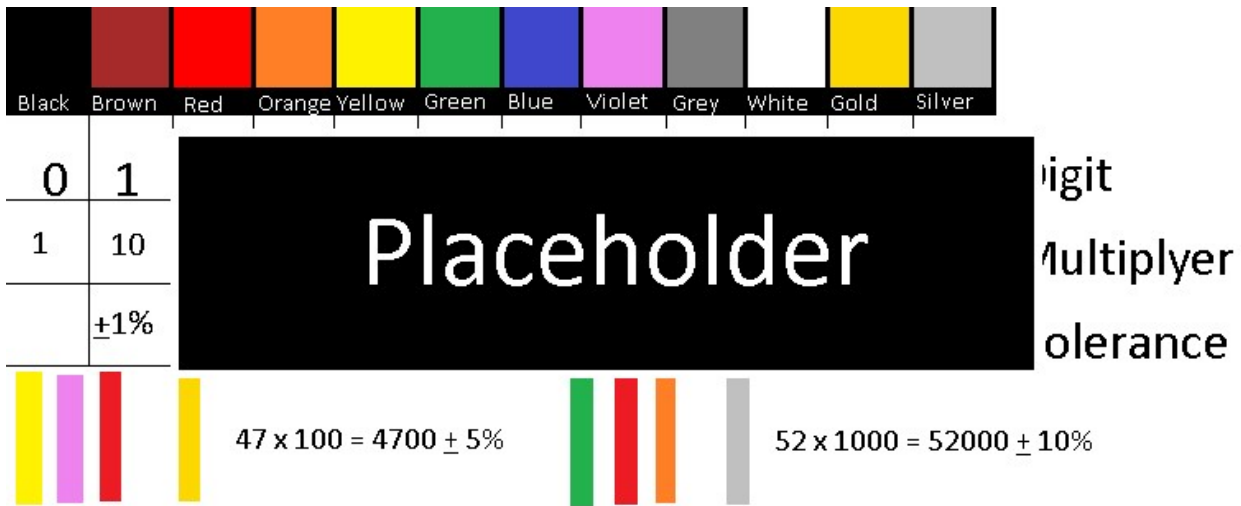
Access points use wifi to connect to the network instead of a hard wired connection such as ethernet. The device has a NIC which in this case is an antenna connected to an IC chip that transmits data. They use a 2.4 GHz or 5 GHz, giga Hertz, or very high frequency[†]. 2.4 GHz is the very close to cellular phones and many other household devices, so 5 GHz is sometimes preferred, unless not available for legacy devices. In most homes and small offices, it is common to see the router using a wifi antenna to act as the access point, but often you will find that the access point is just a device in *bridge mode*, passing traffic to the router like a wireless switch.

Wifi antennas usually cover an area of about 300 ft, give or take a little depending on outside factors like interference, obstacles, material of walls, etc. Since it is possible in some areas to have multiple routers in close proximity and broadcasting on the same frequency, there are 11 20 MHz channels (in North America) to help avoid overlap and the interference that accompanies it. This is one area to look at if you are investigating a signal quality issue on a wifi access point or router. In order to troubleshoot properly it is important to have a basic understanding of how the thing works.

[€] supervisory control and data acquisition - used for industrial control systems.

[†] the actual range for 5GHz is 5.2 - 5.6 GHz, so we say 5 GHz to cover them all.

In this case it is radio waves. Technically sound and light are electromagnetic energy and the frequency is what lets us know if we can see or hear them, or how to classify and use them. Since they are all waves, the frequency we talk about is how fast it *oscillates*, or vibrates, and is represented on a graph in the form of a *sine wave*. Using *Fourier transforms*, mathematical formulas, all electromagnetic energy can be broken down into a sine wave. Visible light has a wavelength of 380nm to 750nm and a frequency of 480 to 790 THz. Please note that the shorter wavelength has a larger frequency. 380nm is about 790 THz while 750nm is closer to 480 THz. If you think about it, it makes sense because the shorter wavelength allows more waves to fit into the same area. Sound has much longer wavelengths, anywhere from about 20 Hz to 20 kHz. The wavelength of a 20Hz sound is about 1.7m while 20 kHz is about 1.7cm. Compare that to the 380nm of visible light on the violet side of the spectrum.



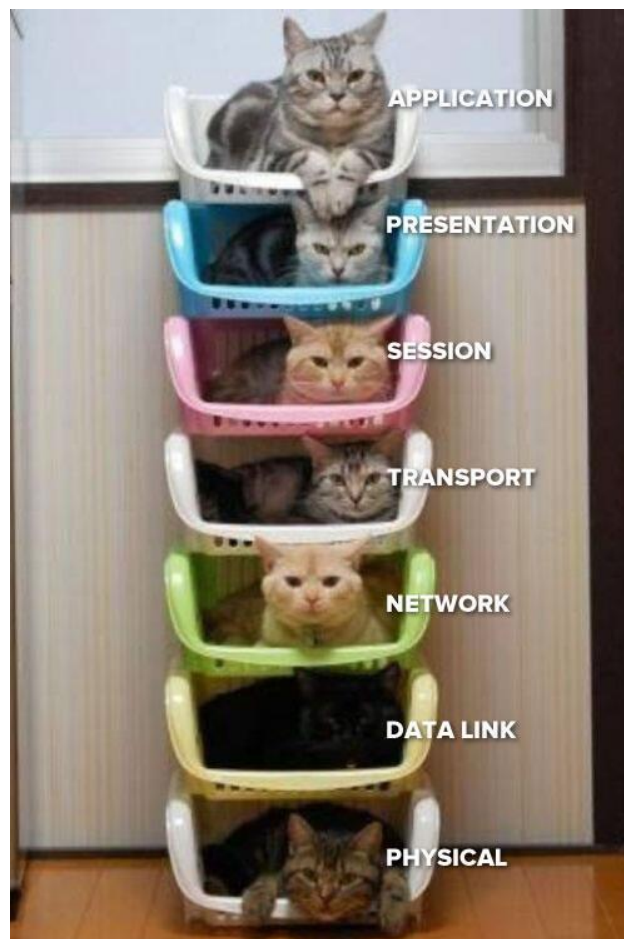
I could write a book on electromagnetic waves and how they interact to produce light and sound, well, I could but the subject is complex enough to keep you busy studying it for a bit, so I'm going to stop there for our purposes of introductory learning. Now that you know that the wifi spectrum includes 2.4 GHz, you can use *Maxwell's equations* to figure out that you can make a loop antenna with a 13.4cm piece of wire to serve as a wifi extender for a place with large obstacles blocking the signal^f. You should be able to figure out why TV signals penetrate things better than wifi. It has to do with the wavelength and the signal strength of transmission. That is a little bit beyond the scope of this book, but knowing that you can find out using a little more research is exactly what this book is intended to do.

One thing you cannot do is read this book and think that's all you have to do. This is just the beginning. You need to continue to learn if you

^f <https://www.makeuseof.com/diy-cell-phone-signal-boosters/>

want to remain competent; maybe not be immersed in it all the time, but you need to keep up with any changes in processes or technology. A good way to do that is find a tech support forum to spend some time on helping folks out. I have a few that I volunteer in and also a few people whose computers I take care of for free, former customers who are older and on fixed incomes but still have computer problems occasionally. I recommend that everyone has to volunteer work they do if for no other reason than to keep your skills sharp on things you may not regularly do.

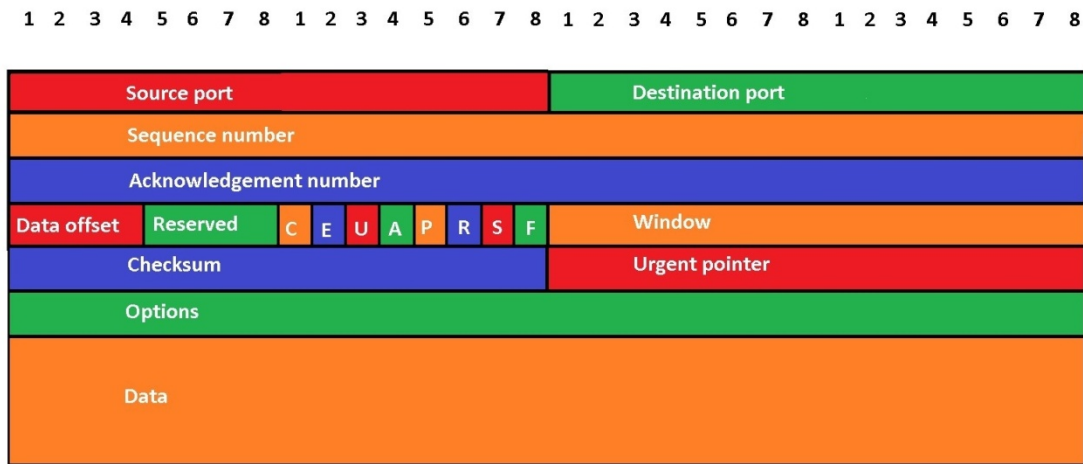
Protocols



Recall that we briefly discussed the OSI model and how it was used to illustrate the various devices interact with your data. Layer 1 encompasses all of the infrastructure such as wires or fiber optic cables, or the physical medium that the data travels on. In some cases it would be radio waves for wifi traffic. Layer 2 and 3 devices are modems,

switches, and routers, or anything that moves the data across the network. Some switches are layer 2 devices and some are layer 3 devices, modems and ONTs are layer 1 devices, and routers are layer 2 devices. The last four layers reside in your computer or workstation.

To make it easier to understand, imagine you are using a web browser to look at cat pictures. When you open the browser, you are using layer 7, the application layer, to act as an interface between you and the network. This is where protocols like HTTP, HTTPS, FTP, etc are used. Layer 6 is the next layer, used primarily for encryption and converting high level languages to low level languages. Layer 5 is where sessions live. For example, when you use TLS, transport layer security, a negotiation takes place where an encryption type and key is agreed on so that your transmission is secure. Then layer 4 is how the data actually travels across the wire. This is the TCP packet from earlier:



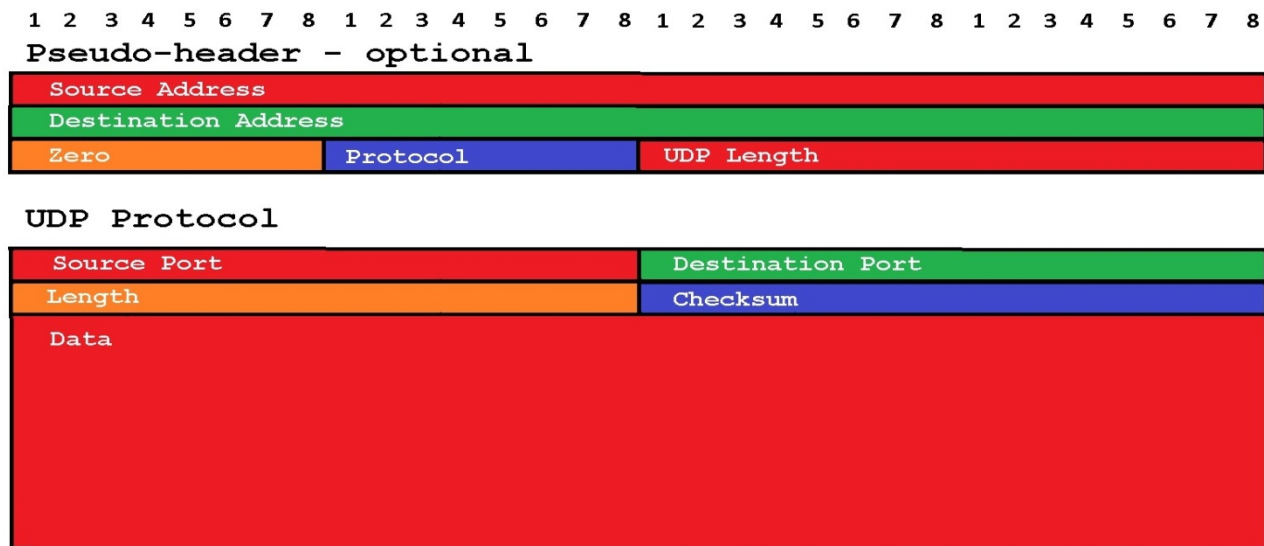
TCP packet structure RFC 9293

- The numbers across the top represent the number of bits.
- The sequence number, acknowledgement number, and options are all 32 bits.
- The source port, destination port, window, checksum and window are all 16 bits.
- The data offset and reserved sections are each 4 bits.
- The flags are all single bits and considered set if they are 1 and not set if they are 0.
- C is CWR - congestion window reduction
- E is ECE - ECN echo - explicit congestion notification
- U is urgent. When this bit is set, the address of the pointer is located in the urgent pointer field.
- A is ACK - acknowledged.
- R is RST - reset
- S is SYN - synchronize
- F is FIN - finish

All of the areas above the Data block at the bottom are added to the packet by the TCP protocol at layer 4 just before it is sent to the network. All of the layers above 4 add their own headers for identification purposes, but each lower layer only sees it as 'data; and not what the data is. When it is passed along back up the line, the previous headers are discarded as the new ones are read. The TCP packet has what is known as a 'three-way handshake' to initiate a secure

connection which allows for transmission reliability. When the connection is made, the first packet has the SYN bit set, the return packet has the SYN and ACK bits set, and the reply to that packet has the ACK bit set. TCP has tracking and is very reliable while UDP on the other hand is not. It has no error correction, no protocol negotiations, and no tracking. It is primarily used in video conferencing and VoIP telephone systems. These systems need the speed of transmission in order to work and have their own error correction capabilities to account for missing packets or out of order receipt of packets.

In the picture below, the pseudo-header is optional and is included in the RFC so I mapped it out for reference even though it is not normally used. This means that UDP has five fields used while TCP uses 19.



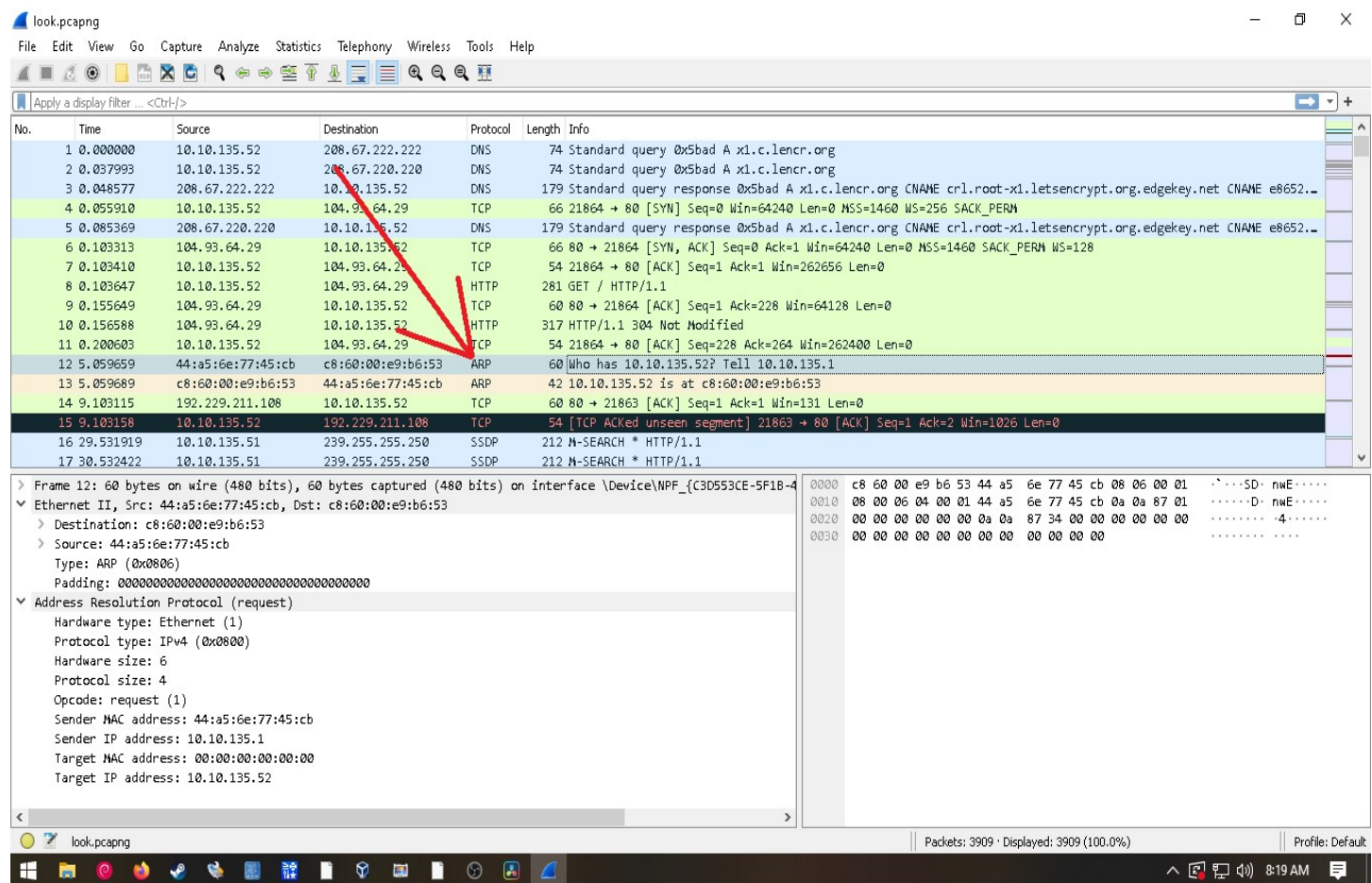
UDP packet structure RFC 768
 The pseudo-header is optional and included for reference only.
 The numbers across the top represent the number of bits.
 The source address and destination address are 32 bits.
 The UDP length, source port, destination port, length, and checksum are 16 bits.
 The zero and protocol are 8 bits.
 The data is variable but 508 bytes is the maximum safe size recommended by some.

So, what does all of this mean to you? If you know how data is supposed to act when travelling and what the different devices have to do with it, when it isn't working properly you should be able to track down where it started going wrong and then look to that device to start troubleshooting. Just a quick example, suppose you open a web browser and can't get to a website. If you know that the browser uses HTTP or HTTPS protocol, you know you can check layer 7 real quick by opening a command prompt, typing `ping yahoo.com`, and then hit enter. This does two things: it checks your DNS by resolving yahoo.com to an IP address and then checks your network connection by sending four 32 bit ICMP packets, internet control message

protocol. If it gets a reply, the network connection is good and it is probably in the browser or somewhere else in the computer. If it doesn't get a reply, the error message can help you determine what part of the process went wrong and where to start troubleshooting. We'll get into further details on that and other troubleshooting techniques later on in the book, that was just a quick example of how to apply what we learned.

Some of the protocols you'll be most interested in are: DNS (it's always DNS) domain name system, ARP address resolution protocol, HTTP hyper text transfer protocol, HTTPS HTTP secure, FTP file transfer protocol, SFTP secure FTP, SSH secure shell, POP post office protocol, IMAP interactive mail access protocol, IPv4 internet protocol version 4, IPv6 internet protocol version 6, LDAP lightweight directory access protocol, and SNMP simple network management protocol. This is by no means an exhaustive list, you can find more here^N, but these are some you should familiarize yourself with.

The following picture shows a pcap (packet capture) using Wireshark, one of the best packet analyzers out there. Notice the arrow pointing to the ARP request. The sections under the colored portion are contents of the packet explained on the left and the data on the right.



^N https://en.wikipedia.org/wiki/List_of_RFCs

It shows the router, 10.10.135.1, asking who has 10.10.135.52. The response is directly under it with 10.10.135.52 responding to 10.10.135.1 announcing the MAC address of the device so the router can verify its routing table. The following picture shows a TCP three way handshake and the encryption negotiation that follows.

The screenshot displays a network capture in Wireshark. The main pane shows a list of packets with the following key entries:

| No. | Time | Source | Destination | Protocol | Length | Info |
|-----|-----------|----------------|--------------|----------|--------|---|
| 22 | 35.710047 | 208.67.222.222 | 10.10.135.52 | DNS | 226 | Standard query response 0xd1c6 A v10.events.data.microsoft.com CNAME win-global-asimov-leafs-events-data.t... |
| 23 | 35.711326 | 10.10.135.52 | 20.42.65.89 | TCP | 66 | 21865 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 WS=256 SACK_PERM |
| 24 | 35.728098 | 208.67.220.220 | 10.10.135.52 | DNS | 229 | Standard query response 0xd1c6 global-asimov-leafs-events-data.t... |
| 25 | 35.742040 | 20.42.65.89 | 10.10.135.52 | TCP | 66 | 443 → 21865 [SYN, ACK] Seq=0 |
| 26 | 35.742140 | 10.10.135.52 | 20.42.65.89 | TCP | 54 | 21865 → 443 [ACK] Seq=1 Ack=1 Win=263424 Len=0 |
| 27 | 35.742805 | 10.10.135.52 | 20.42.65.89 | TLSv1.2 | 268 | Client Hello |
| 28 | 35.772936 | 20.42.65.89 | 10.10.135.52 | TCP | 1514 | 443 → 21865 [ACK] Seq=1 |
| 29 | 35.773896 | 20.42.65.89 | 10.10.135.52 | TCP | 1514 | 443 → 21865 [ACK] Seq=1461 Ack=215 Win=525312 Len=1460 [TCP segment of a reassembled PDU] |
| 30 | 35.773896 | 20.42.65.89 | 10.10.135.52 | TCP | 1514 | 443 → 21865 [ACK] Seq=2921 Ack=215 Win=525312 Len=1460 [TCP segment of a reassembled PDU] |
| 31 | 35.773896 | 20.42.65.89 | 10.10.135.52 | TLSv1.2 | 76 | Server Hello, Certificate, Server Key Exchange, Server Hello Done |
| 32 | 35.773896 | 10.10.135.52 | 20.42.65.89 | TCP | 54 | 21865 → 443 [ACK] Seq=215 Ack=4403 Win=263424 Len=0 |
| 33 | 35.773896 | 10.10.135.52 | 20.42.65.89 | TLSv1.2 | 212 | Client Key Exchange, Change Cipher Spec, Encrypted Handshake Message |
| 34 | 35.773896 | 20.42.65.89 | 10.10.135.52 | TLSv1.2 | 105 | Change Cipher Spec, Encrypted Handshake Message |
| 35 | 35.773896 | 10.10.135.52 | 20.42.65.89 | TLSv1.2 | 1052 | Application Data |
| 36 | 35.813769 | 10.10.135.52 | 20.42.65.89 | TLSv1.2 | 982 | Application Data |
| 37 | 35.842509 | 20.42.65.89 | 10.10.135.52 | TCP | 60 | 443 → 21865 [ACK] Seq=4454 Ack=2299 Win=525568 Len=0 |
| 38 | 35.844066 | 20.42.65.89 | 10.10.135.52 | TLSv1.2 | 506 | Application Data |

The detailed view of Frame 23 (Transmission Control Protocol) shows the following fields:

- Source Port: 21865
- Destination Port: 443
- [Stream index: 2]
- [Conversation completeness: Complete, WITH_DATA (31)]
- [TCP Segment Len: 0]
- Sequence Number: 0 (relative sequence number)
- Sequence Number (raw): 417029245
- [Next Sequence Number: 1 (relative sequence number)]
- Acknowledgment Number: 0
- Acknowledgment number (raw): 0
- 1000 ... = Header Length: 32 bytes (8)
- Flags: 0x002 (SYN)
- Window: 64240

Wireshark is one of the best tools for troubleshooting network traffic there is. It is available as a free download at <https://wireshark.org>. We will cover it more thoroughly in the network troubleshooting section.

Cloud and Virtualization

The Cloud

You may have heard the phrase, "the cloud is just someone else's computer". This is true in a sense. Usually it's a data center with thousands of servers, each running hundreds of VMs, or virtual machines. But it's still just a computer in the end. Enterprises are moving away from onsite servers in a lot of cases and saving money by renting the services. Google and Amazon are two of the largest that I know of, but there are others.

Cloud computing comes in three types: Public, private, and hybrid. Public clouds share resources among clients, private clouds restrict access to resources local to each company, and hybrid clouds are a mixture of the two. Some industries require private clouds for regulatory purposes such as banking or credit card use.

As a tech, the part you will be concerned about is how to access them to work on them. Usually they have a dashboard for access via an internet browser, although there are other methods such as remote desktop. As a level one tech, you probably won't have much access on a production server. When you do work on the, remember that it is just another computer, and all of the same troubleshooting steps apply, but you have to be careful about losing remote access. Always make sure you have backed up your data when possible before doing anything that could cause you to lose it. And if you don't test your backups regularly, they're not backups. Since servers are outside the scope of this book this is about as far as we're going to go with cloud computing.

Virtualization

Virtualization is running a computer image as a separate operating system on top of an already running operating system, or host, sharing the same hardware. One computer with enough resources such as CPU cores and RAM can run multiple virtual machines. Each virtual machine acts as an independent computer. It makes deployments of new machines quick and easy since you can just clone the machine as many times as you need. If you get a virus or corruption, you just delete it and copy another clone.

When you run a virtual machine, you should use a sandbox, or type of virtual cage, that helps prevent malware from infecting your host computer. There are a few different virtualization programs that all act pretty much the same. Since a virtual machine is just another computer and each program has different features and controls, I'm not going to go

into much more detail than that. Virtual environments are just like your own working environment sandboxed and recreated with minimal resources. This allows you to quickly recreate them in case of corruption.

Containers are another type of virtualization that is becoming more popular. They are a small code block that runs an isolated computer environment that contains everything needed to run applications. They act independently of one another and the host so that you can run many of them at the same time. Docker images are one example of containers that are pretty popular.

Appendix 1

Windows System Utilities Run Commands

| Command | What It Does |
|--------------|--|
| command | opens the command prompt |
| compmgmt.msc | opens the computer management console |
| devmgmt.msc | opens the device manager |
| diskmgmt.msc | opens the disk management tool |
| eventvwr.msc | opens the event viewer |
| fonts | opens the fonts folder |
| fsmgmt.msc | opens shared folders |
| gpedit.msc | opens the group policy editor |
| lusrmgr.msc | opens the local users and groups |
| mailto: | opens the default mail client |
| msconfig | opens the system configuration utility |
| msinfo32 | opens the system information utility |
| perfmon.msc | opens the performance monitor |
| resmon | opens the resource monitor |
| regedit | opens the registry editor |
| rsop.msc | opens resultant set of policy |
| secpol.msc | opens local security settings |
| services.msc | opens services utility |
| system.ini | Windows loading information |
| win.ini | Windows loading information |
| winver | shows the current version of Windows |

Control Panel Commands

| Command | What It Does |
|-------------------|-----------------------|
| appwiz.cpl | add/remove programs |
| timedate.cpl | time/date properties |
| desk.cpl | display properties |
| inetcpl.cpl | internet properties |
| main.cpl keyboard | keyboard properties |
| main.cpl | mouse properties |
| mmsys.cpl | multimedia properties |
| mmsys.cpl sounds | sound properties |
| sysdm.cpl | system properties |

License

This work is created and distributed under [Creative Commons Attribution-Non Commercial 4.0 International Public License](https://creativecommons.org/licenses/by-nc/4.0/).

You are free to share – copy and redistribute the material in any medium or format – and adapt – remix, transform, and build upon the material –under the following terms:

Attribution – You must give appropriate credit, provide a link to the license, and indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

Non commercial – You may not use the material for commercial purposes.

No additional restrictions – You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

The licensor cannot revoke these freedoms as long as you follow the license terms.

This license only covers the original work that belongs to me including the lab. Not all of the course material is covered under the same license; some of the work is being used with written permission from the authors. Check the license on all material before using.