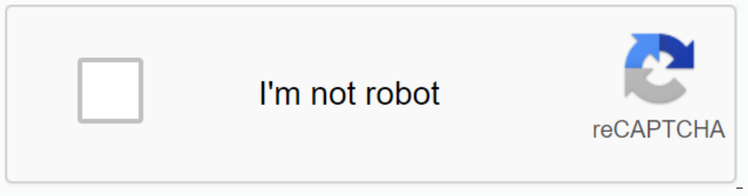

Continue

Check open ports windows 7

In another article, we explained the computer ports and what they are used for. Apart from that, what can we do with port information? Since all the in-and-out traffic on your computer passes through the ports, we can check them to see what they are doing. Maybe the port isn't listening to traffic? Maybe something is using a port that shouldn't be? We will use the cleansed Windows command to view our listening ports and PID (Process ID). We'll also see what we can do with this information. What is Netstat? The netstat command is a combination of the words 'network' and 'statistics'. The netstat command works on all versions of Windows from Windows XP to Windows 10. It is also used on other operating systems (SS) such as Unix and Linux, but we will stay with Windows here. Netstat can provide us with: The name of the protocol used by the port (TCP or UDP). The local IP address and the name of the computer and the port number being used. The IP address and the port number to which we connect. The status of a TCP connection. For more information about what these states are, read rfc 793 Event Processing. Using Netstat To View Listening Ports && PID Use win key combination + X. In the menu that opens, select Command Prompt. Enter the <pre>netstat -a -n -o</pre>. The parameters for netstat are preceded by a hype, not a slash like many other commands. The -a tells you to show us all the active connections and ports that your computer is listening to. The -n tells netstat to show only IP addresses and ports as numbers. We're telling him not to try to sort out the names. This makes for a faster and cleaner screen. The -o tells netstat to include the PID. We will use the PID later to find out which process is using a specific port. View the results and take note of addresses, port numbers, status, and PID. Let's say we want to know what the port 63240 is using. Note that your PID is 8552 and is connecting to ip address 172.217.12.138 to port 443. What is using this port? Open Task Manager. This is done more easily using the ctrl + Shift + Esc key combination. Click the Details tab. To make this easier, click the header of the PID column to sort pids numerically. Scroll down to PID 8552 and see what the process is. In this case, it is googledrivesync.exe. But is it really? Sometimes viruses can become legitimate processes. In a web browser, go to ipinfo.io. Enter ip address 172.217.12.138. As we can see, the IP address is registered with Google. So this googledrivesync.exe is legitimate. How to get port, PID, && Process Name In PowerShell PowerShell is the newest way to to use a command line interface with Windows. We say newer, but it's been around several versions. You should learn PowerShell even if you're a home user. Most Windows Windows they also work on PowerShell, we can also combine them with command-command-lets pronounced PowerShell cmdlets. Joe Winteltools.com provides the script for this method. Open Notepad and enter the following code: $netstat = netstat -aon | Select -String -pattern (TCP| UDP) $processList = Get-Process foreach($result to $netstat) { $splitArray = $result -split $procID = $splitArray [$splitArray.length – 1] $processName = $processList | On-Object {$_.id -eq $procID} | select the process name $splitArray[$splitArray.length – 1] = $procID + $processName.processname $splitArray -join } Save file as get-NetstatProcessName.ps1. Be sure to keep in mind where you are saving. It is important to change the Save as Type: in All Files (*.*) or it will be saved as get-NetstatProcessName.ps1.txt and it will not work for us. Open PowerShell and go to the location where the script was saved. In this case, it is <pre>cd C:\Scripts Cd (Cd C:\Scripts) Cd C:\Script</pre>. Hit Enter to run the order. Run the script using dot-sourcing to make it work. This means using ./ before the file name. The command will be <pre>./get-NetstatProcessName.ps1 (English)</pre>Now we can see all traditional netstat information more the name of the process. There is no need to open Task Manager anymore. Go find them We have covered two ways to use the netstat command to view the listening ports. It can be used either on the old command prompt or within a PowerShell script. With the information you can give us, we have looked at how you can help us figure out what our computer is doing. If you think netstat is a great utility, take a look at other Windows TCP/IP utilities, such as tracert, ipconfig and nslookup. Or use Resource Monitor to get a better look at the hidden website and Internet connections. There's a lot you can do to see exactly what your computer is doing. Have you used netstat to solve a problem? Please tell us what you did. Any questions about how to use netstat? Please ask us in the comments below. There is a good chance it has happened in this article because an application that is trying to run complains about a port that crashes or has read about how leaving certain ports open on your network can be a security issue. Either way, at the end of this piece you will not only know what these ports everyone is going through are, but how to check your computer to find open or closed ports. What is a network port? The first thing you need to know is that the ports we refer to here are virtual. It has nothing to do with physical network hardware ports on your router, TV, consoles or computers. Ports are simply a way for network hardware and software organised information traffic. Think of the reserved lanes on a road. The sidewalk is for pedestrians. There may be a dedicated bike lane. Carpool's vehicles and buses also have their own lanes. The ports are used to Function. One port can be used to receive emails, while another brings file transfer requests or website traffic. There are two common types of ports, which need a brief explanation before going on to check which ports on your system are open and which are not. What are TCP and UDP ports? The two common types of ports in modern networks are known as TCP and UDP ports. That is, transmission control protocol and user datagram protocol respectively. So these two types of port use different network protocols. You may think of them as distinctive sets of rules for how to send and receive bits of information. Both types of port are built on the fundamental Internet Protocol (IP) that makes internet and home networks, well, work. However, they are suitable for different applications. The big difference is that when you send information about UDP, the sender does not first have to establish a connection with the receiver before starting the conversation. It's a bit like sending a letter. You do not know if the other person received your message and you have no guarantee that you will receive any comments. TCP, on the other hand, is more like making a phone call. The receiver must pick up the connection and there is a flow of information back and forth until someone deliberately hangs up. UDP messages are generally broadcast over a network to anyone who is listening to the specified UDP port. This makes it perfect for cleaning type messages that refer to the execution of the network itself. It is also perfect for voice streaming over IP, online video games and streaming broadcasts. What? These applications benefit from the low latency of UDP and the constant flow of information that does not have to be perfect to be useful. Some corruption in your Skype chat is much less important than the low amounts of delay, after all. TCP is much more common than UDP and absolutely ensures that all data is received free of errors. Almost everything that doesn't need the specific advantages of UDP, uses TCP instead. Which ports usually open by default? There are a lot of ports. A port number can be anything from 0 to 65535! That doesn't mean any application can only choose any port. There are established standards and ranges, which helps us to make sense of the noise. Ports 0-1023 are associated with some of the most important and fundamental network services. This makes sense, since smaller ports were first assigned. The SMTP protocol for e-mail, for example, is used exclusively by port 25. Ports 1024-49151 are known as registered ports and are assigned to important common services such as OpenVPN on port 1194 or Microsoft SQL on ports 1433 and 1434. The rest of the port numbers are known as dynamic or private sectors. These ports are not reserved and anyone can use them on a network to support a The only problem arises when two or more services on the same network are using the same port. While it is impossible to list all important ports, these common ports are useful for meeting memory: Since there are so many thousands of common port numbers, the easiest approach is to remember the ranges. Which will tell you whether a given port is reserved or not. Thanks to Google, you can also search which services use a specific port at any time at all. Find open ports in Windows Now that we have all the basic knowledge about the TCP and UDP ports out of the way, it's time to get down to the process of finding which ports are open and in use on your computer. The good news is that Windows has a pretty useful command built into it that will show you which ports are currently being used on your computer by various applications and services. The first thing you want to do is open the Start menu and search for CMD. Now, right click on CMD and Run as Administrator. With the command prompt open, type: Netstat -ab Don't worry about a long list of information scrolling faster than you can read it. You can simply use CTRL+C and CTRL+V to copy and paste the information into notepad or any other text editor. The information in brackets is the name of the program that uses the port. TCP or UDP refers to the protocol used in this port. The number consists of an IP address, and then the port number after the two points. Scan for blocked ports that is responsible for finding which ports are being used and by which application, but does not tell us which ports are being actively blocked by the Windows Firewall. Again, open the Start menu and search for CMD. Right-click on CMD and Run as Administrator.With the Command Prompt open, type: netsh firewall demo state This is a blocked and open port display according to the settings of your Windows Firewall. You'll see a note about this outdated order, but the new command doesn't show us the information we want. So for now the use of the 'show state' command remains the fastest and easiest way to obtain port information. Just because the Windows Firewall is not blocking a port, that doesn't mean your router or ISP are not. So the last thing we want to do is check if some external blocking is occurring. Open the Start menu and search for CMD. Now, right-click CMD and Run as Administrator. With the command prompt open, type: netstat -ano | findstr -i SYN_SENT If you don't get any list hits, then nothing is being blocked. If some ports are listed, they mean they are being blocked. If a port not blocked by Windows appears here, you may want to check your router or send an email to your ISP, if you switch to different port is not an option. Useful applications to trace the status of the port While the command prompt is a good quick and dirty tool, there are more refined third-party applications than will help you get an image of the port settings. The two featured here are just popular examples. SolarWinds Free Port Scanner SolarWinds requires you to send your name and data to download it, but you're worried if you put your actual information on the form or not. We tested several free tools before we settled on SolarWinds, but it was the only tool that both worked properly under Windows 10 and had an easy interface. He was also the only one who did not activate a false positive virus flag. One of the big problems with port scanning software is that security companies tend to see them as malware. So most users ignore any warning of viruses that come with such tools. This is a problem because you can't tell the difference between a false positive and a real virus in these applications. SolarWinds might come with some strings attached, but it actually works as advertised and is easy to use. CanYouSeeMe This is, as you can probably say, a website service instead of an app. It is a good first port of scale to see if external data can pass through the local port or not. Automatically detect your IP address and all you have to do is specify which port to try. It will then tell you whether the port is locked or not and then you will have to find out if the obstruction is on your computer, router, or service provider level. Conclusion For most users, ports are not something you need to worry about. They are managed by your operating system, applications, and network hardware. When things go wrong, however, it's good to have the tool in hand that allow you to find open ports to smell suspicious activity or find out where exactly your precious information is hitting a brick wall. Wall.