

## GT New Horizons

# Lag

---

**Lag** or latency is a data transfer problem between a source and destination. This page will help you to understand it better and to diagnose it. Lag has multiple causes including ping issues, packet loss, CPU Load (TPS) and rarely, coding bugs.

## Poor FPS is Not Lag

---

FPS (frames per second) is a separate issue and never has anything to do with a network. Low FPS a hardware or settings problem - graphics card, RAM or CPU is not powerful enough to keep up with the game's graphical demand or a configuration is artificially limiting FPS. In Minecraft, a FPS cap can be set under ESC > Options > Video Settings > Max Framerate. Sometimes the same root cause can result in both lag and FPS drop.

## Ping Issues

---

Ping is a metric measuring how long it takes for a network packet to go from a computer to the server, or vice versa. It is expressed in milliseconds (ms), and the lower it is, the better & faster it's communicating with the server. (Or you are getting closer if the propagating medium didn't change).

Before digging more into networking stuff, let's talk about how the internet is structured as a whole. It's mainly composed of:

- routers (or nodes)
- endpoints (servers or clients)
- connections between them (links). The place where you plug a link to a router or an endpoint is called an interface.

A sub network of internet with the same policy for routing is called an Autonomous System (AS).

A ping measure follows these steps:

1. measuring the time just before sending a "hello" packet
2. sending a "hello" packet to the server
3. waiting for a "hello" packet from the server
4. measuring the time at the arrival of the packet
5. computing the delta and dividing it by 2 (because of the round trip)

Inside a router, here are the following source of delays

1. delay of arrival (time of propagation through the medium in the input link)
2. delay of buffering in the input buffer (fixed by hardware)
3. delay of processing (influenced by the global load of the router, such as other services or other

X: 348 Y: 173

us deliver our services. By using our services, you  
use of cookies.

**More information**

**OK**

### 5. delay of departure (time of propagation through the medium in the output link)

Also, a router is supposed to find the fastest path for routing your packets. For that, some routing algorithms are used, and usually, the route with the least amount of hops is preferred (a hop is a jump to one node to another). The issue is that between 2 AS, there can be what is called an administrative distance. It's simply a faked metric for routing algorithms, and it can impact greatly the routing. It is done by companies or governments for various reasons. Due to that, the fastest route will sometimes appear as the slowest route, causing your traffic to go through a longer route, increasing your ping.

There is also the fact that routers can sometimes become unavailable (broken link, overloaded router, power failure, etc). When a router did not contacted its neighbors after some times, the neighbors will remove the faulty router from their routing table, which causes your route to be deviated.

To sum thing up, your route is evolving over time, and can be impacted, independently from you, the server, or your ISP.

## Packet loss issues

---

To convey your network packets over internet, there are two big protocols encapsulating your data: TCP and UDP. TCP is used when an application must transmit data with integrity (packet ordering, re-emission of lost packets, etc), and UDP is used when an application must transmit data as fast as possible (streaming for instance). Minecraft uses TCP, because the game requires to process correct information and in the correct order to synchronize the client with the server.

A packet can be lost because of external reasons (router going down, etc), but also on purpose by the routers. In networking, nodes must be as fast as possible, so this translates into doing the least work possible. If the input buffer or the output buffer is full but a packet is incoming, instead of waiting, or asking for a resend of the packet to the previous router, the router will do the least work possible, which is trashing the packet. The integrity checks are delegated to client and server.

Packet loss does not impact your ping per se, but a too high rate of packet loss will cause you to feel a bigger latency than what the ping displays. And this is normal. Let's assume a ping to a server of 50 ms. The client initiate a TCP connection with the server, and start sending data to the server. After each block of data sent, the client will ask if the server got them, if not, the client re-emit the packet, and ask again if the server got it. The client will keep emitting the packet until the server tells the client it got it. But each time, the data took 50 ms to be transmitted, so the higher is the amount of re-emission, the bigger will be the difference between the latency displayed by the ping and the perceived latency in game.

In short, packet loss can increase your latency, no matter how good your ping is.

## CPU Load Issues

---

X: 348 Y: 173

us deliver our services. By using our services, you  
use of cookies.

**More information**

The CPU load on the server can be responsible for some latency. To understand it, let's talk about the unit time used internally by Minecraft: the tick. The game is designed in such way that 1 second corresponds to 20 ticks. So a tick is corresponding to 0.05s, or 50 ms. So each tick cycle, the game computes what needs to be computed within those 50ms. If it is faster than 50 ms, the game will idle until it reaches 50 ms. What's annoying is when the game needs to compute during more than 50ms to finish the tick. It causes the game to run less ticks per second (TPS). This is an handy metric to discuss the lag of the server, but practically the tick time is measured in ms, then a formula is used to convert it in term of TPS:

$$TPS(x) = \begin{cases} 20 & \text{if } x < 50 \text{ ms} \\ 20 \times \frac{50}{x}, & \text{if } x \geq 50 \text{ ms} \end{cases}$$

So even with the best connection in the world, if your server is slowing down, you'll feel the lag. For large packs such as GT:NH, lag is unavoidable, the more you progress, the more you need to learn about tricks to avoid lag. Devs do what they can to mitigate the issue, but it's not easy to just get rid of the unnecessary CPU load.

## Bugs

---

Network handling in mods can sometimes cause desync or artificial latency. If you made sure it's none of the sources of lag above, feel free to open a ticket on the GT:NH bug tracker, we will gladly help you to hunt down the bug.

## Troubleshooting the lag sources

---

### Ping and Packet loss

#### Command line tools

- To monitor the ping, you can open terminal and use the following command `ping -t <your server IP or domain name>`. This will continuously check send ping requests to the server.
- To get the detailed route your packets are following to reach the server, run `tracert <your server IP or domain name>` on windows, or `traceroute <your server IP or domain name>` This will give you the nodes your traffic is going through, as well as your ping between you and them during the traceroute. Handy to find if there is a troublesome node ruining your traffic. If you see some nodes being named \* it's normal, some nodes are configured to ignore ping requests (what does the traceroute behind the scenes) for security or performance reasons.
- on Linux, you can install MTR, for Matt's TraceRoute, it's a tool being a bit more precise than ping+traceroute combo, as it also gives the packet loss rate to each node. to use it, simply run `mtr <your server ip or domain name>`

#### GUI software

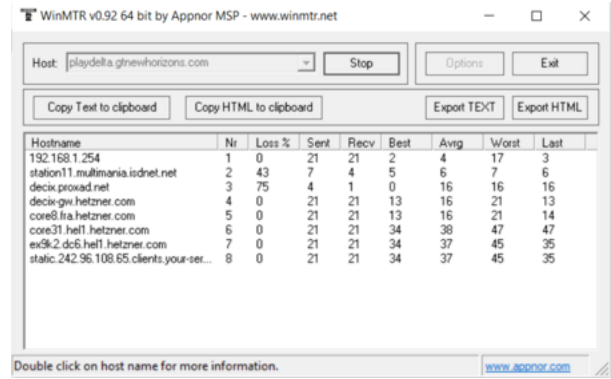
- On windows you can use WinMTR it's basically an interface for the MTR program. Nodes

X: 348 Y: 173

us deliver our services. By using our services, you use of cookies.

**More information**

- VPNs can help you avoid troublesome nodes by being rerouted to the exit point of the VPN before trying to reach the server. This can work, but you need to monitor your connection with a MTR like software, and keep switching to other VPN endpoints until you find one. This technique is advised if you have a very low ping by default, as the VPN increases the length of the route, which increases the ping. It can be worth to have a higher ping but with no packet loss, as regular packet loss is worth than high ping. **There is a flaw to this method: if the faulty node is too close to you or to the server, no matter what VPN you use, you'll end up going through the faulty node.**



WinMTR v0.92 64 bit by Appnor MSP - www.winmtr.net

Host: playdelta.gtnewhorizons.com [Stop] [Options] [Exit]

[Copy Text to clipboard] [Copy HTML to clipboard] [Export TEXT] [Export HTML]

Hostname	Nr	Loss %	Sent	Recv	Best	Avg	Worst	Last
192.168.1.254	1	0	21	21	2	4	17	3
station11.munichia.isdn.net	2	43	7	4	5	6	7	6
decix.proxad.net	3	75	4	1	0	16	16	16
decix-gw.hetzner.com	4	0	21	21	13	16	21	13
core8.fra.hetzner.com	5	0	21	21	13	16	21	14
core31.fra.hetzner.com	6	0	21	21	34	38	47	47
ex9k2.do5.hell.hetzner.com	7	0	21	21	34	37	45	35
static.242.96.108.65.clients.your-ser...	8	0	21	21	34	37	45	35

Double click on host name for more information. [www.appnor.com](http://www.appnor.com)

The interface of WinMTR

## Server load

For tracking down the server load, you can use tools such as Opis to profile stuff in game, such as which dimension is lagging, how many entities there are, which chunks are heavily loaded, what block is generating lag, etc. You can also use profilers such as Warmroast or VisualVM, which will display what part of the code is executed the most.

[v · t · e \(https://wiki.gtnewhorizons.com/wiki/Template:Navbar\\_GTNH?action=edit\)](https://wiki.gtnewhorizons.com/wiki/Template:Navbar_GTNH?action=edit)

[\[Collapse\]](#)

### GT New Horizons



<b>Meta</b>	<a href="#">Acronyms</a> • <a href="#">Discord Commands</a> • <a href="#">Modlist</a> • <a href="#">Official Servers</a> • <a href="#">Server List</a> • <a href="#">Quick Links</a>
<b>Tiers</b>	<a href="#">Beginner Tips</a> • <a href="#">Stone</a> • <a href="#">Steam</a> • <a href="#">LV</a> • <a href="#">MV</a> • <a href="#">HV</a> • <a href="#">EV</a> • <a href="#">IV</a> • <a href="#">LuV</a> • <a href="#">ZPM</a> • <a href="#">UV</a> • <a href="#">UHV</a> • <a href="#">UEV</a> • <a href="#">UIV</a> • <a href="#">UMV</a> • <a href="#">UXV</a> • <a href="#">MAX</a>
<b>Mechanics</b>	<a href="#">Bees</a> • <a href="#">Biome</a> • <a href="#">Electricity</a> • <a href="#">Electricity for Beginners</a> • <a href="#">Guides</a> • <a href="#">Hazard</a> • <a href="#">IC2 Crops</a> • <a href="#">JourneyMap</a> • <a href="#">Loot</a> • <a href="#">Multiblock</a> • <a href="#">NEI</a> • <a href="#">Oil</a> • <a href="#">Ore Generation</a> • <a href="#">Pollution</a> • <a href="#">Power Generation</a> • <a href="#">Quest Book</a> • <a href="#">Tinkers Tools</a> • <a href="#">Waila</a>
<b>Setup</b>	<a href="#">Additional Mods</a> • <a href="#">Backups and Recovery</a> • <a href="#">Commands and Configurations</a> • <a href="#">Debugging</a> • <a href="#">Release Stability</a> • <a href="#">Installing and Migrating</a> • <a href="#">Lag</a> • <a href="#">Low End PCs</a> • <a href="#">Server Setup</a> • <a href="#">Resource Packs</a> • <a href="#">Shaders</a> • <a href="#">World Generation</a> • <a href="#">World Seeds</a>
<b>Development</b>	<a href="#">Pollution Revamp</a> • <a href="#">Development</a> • <a href="#">Ore Spreadsheet Export</a> • <a href="#">Salt Nerf Options</a> • <a href="#">Technical Oregon Information</a> • <a href="#">Upcoming Features</a>

Retrieved from "<https://wiki.gtnewhorizons.com/wiki/Lag?oldid=9726>"

X: 348 Y: 173

us deliver our services. By using our services, you  
use of cookies.

**More information**