Nexthosting

Studie · Gameservere

Serverbelastningstest: Hvor mange protokol-bots kan Vanilla, Paper, Purpur og Fabric med Lithium bære?

Fire serverprogrammer til Minecraft og tre JVM-profiler, målt med protokol-bots på en test-VM

IKKE ET OFFICIELT MINECRAFT-PRODUKT. IKKE GODKENDT AF ELLER TILKNYTTET MOJANG ELLER MICROSOFT.

Studiens type	Egen måling
Udgiver	Nexthosting
Forfatterskab	Erik Friedl (ORCID: https://orcid.org/0009-0008-2784-4314), Nexthosting
Version	1.0.0 fra 22.11.2026
Periode	21.11.2026 til 22.11.2026
Stikprøve	18 af 18 Målekørsler evalueret
Licens	CC BY 4.0
Online-version	nexthosting.net/da/undersoegelser/server-haertetest-minecraft

Indhold

1	Hovedresultater	3
2	Sammenfatning	4
3	Resultater	4
4	Metode	10
5	Stikprøve og periode	12
6	Begrænsninger i, hvad studien kan vise	13
7	Data og kontrollerbarhed	14
8	Kilder	18
9	Åbenhed	19
10	Sådan citerer du denne studie	19
11	Ændringer og rettelser	20

1 Hovedresultater

80 bots

Paper med Aikar-flags

Paper forblev i gennemsnit over målevinduet i alle tre gentagelser op til 80 bots under den på forhånd fastlagte grænse på 50 ms p95-MSPT, i en af dem op til 85. Med mindst 19,5 TPS i gennemsnit som strengere grænse var det 80 i alle tre gentagelser.

Dokumentation: laeufe.csv, Felt kapazitaet · laeufe.csv, Felt kapazitaet_tps · Kilde [1]

75 bots

Purpur, ved Papers kapacitet ikke til at skille ad

Purpur nåede 75 til 80 bots. Intervallet overlapper med Papers (80 til 85), derfor udviser denne måling ingen forskel mellem de to.

Dokumentation: laeufe.csv, Felt kapazitaet

65 bots

Fabric med Lithium

Fabric med Lithium nåede 65 til 70 bots. Det ligger over Vanilla og under Paper og Purpur; intervallerne overlapper i begge retninger ikke.

Dokumentation: laeufe.csv, Felt kapazitaet

55 bots

Vanilla

Vanilla forblev i gennemsnit over målevinduet i alle tre gentagelser op til 55 bots under 50 ms p95-MSPT. Med mindst 19,5 TPS i gennemsnit over målevinduet var det kun 50.

Dokumentation: laeufe.csv, Felt kapazitaet · laeufe.csv, Felt kapazitaet_tps

16,31 TPS

MSPT-grænsen alene er ikke nok

Ved 55 bots lå Vanilla i gennemsnit over målevinduet under 50 ms p95-MSPT, men kørte kun med 16,31 til 17,16 i stedet for 20 ticks pr. sekund. Derfor nævner denne undersøgelse ved siden af den på forhånd fastlagte kapacitet også kapaciteten ved mindst 19,5 TPS.

Dokumentation: daten.csv, Felt tps · Kilde [2]

80 bots

Standard-G1 og Aikar-flags: ingen påvist kapacitetsforskel

Paper med standard-G1 nåede 80 til 85 bots, med Aikar-flags ligeledes 80 til 85. Ved kapaciteten er der ingen påvist forskel. Supplerende observeret: Ved p99-værdien lå Aikar-flags i 7 af 13 fælles trin adskilt lavere, standard-G1 i ingen. Aikar-kørslerne lå i gennemsnit tidligere i rækkefølgen, derfor kan en del af afstanden stamme fra driften over målenatten (se begrænsninger).

Dokumentation: laeufe.csv, Felt kapazitaet · p99-aikar-g1.csv, Felt spieler · Kilde [1]

70 bots	**Generational ZGC lå lavere** Paper med Generational ZGC nåede i alle tre gentagelser 70 bots og dermed færre end med standard-G1 eller Aikar-flags (hver 80 til 85). Det gælder for denne belastningsprofil, denne heap og denne Java-version. Dokumentation: laeufe.csv, Felt kapazitaet · Kilde [3]
18 af 18 kørsler	**Alle kørsler gyldige** Alle 18 kørsler opfyldte de på forhånd fastlagte regler. I gennemsnit over et trin nåede CPU-steal højst 0,061 %, belastningsgeneratorens kerner højst 5,72 %. Dokumentation: laeufe.csv, Felt lauf · daten.csv, Felt steal_pct · daten.csv, Felt cpu_bots_pct

2 Sammenfatning

> **Problemstilling:** Hvor mange samtidige protokol-bots kan Vanilla, Paper, Purpur og Fabric med Lithium klare på samme hardware, og hvor meget påvirker JVM-indstillingerne resultatet?

Vi har belastet Vanilla, Paper, Purpur og Fabric med Lithium i Minecraft 26.3 på samme virtuelle maskine. Protokol-bots sluttede sig til trinvist, indtil serveren var overbelastet. Alle 18 kørsler var gyldige efter de på forhånd fastlagte regler.

Som kapacitet tæller det højeste antal bots, op til hvilket p95-ticktiden i gennemsnit over målevinduet forblev under 50 ms, og det i alle tre gentagelser. Paper nåede 80, Purpur 75, Fabric med Lithium 65 og Vanilla 55 bots. Paper og Purpur kan ikke skilles ad ved den på forhånd fastlagte kapacitet; for alle andre programpar overlapper gentagelsernes intervaller ikke.

Ved JVM-indstillingerne viste der sig på Paper ingen påvist kapacitetsforskel mellem standard-G1 og Aikar-flags. Generational ZGC lå med 70 bots lavere.

Ticktiden alene viser ikke enhver overbelastning: Vanilla forblev ved 55 bots i gennemsnit under 50 ms p95, men nåede kun 16,31 til 17,16 i stedet for 20 ticks pr. sekund. Derfor står der ved siden af hver kapacitet også værdien ved mindst 19,5 TPS i gennemsnit. Tallene gælder for denne VM og denne bot-profil og kan ikke overføres direkte til rigtige spillere.

3 Resultater

Sådan læser man resultaterne

Den på forhånd fastlagte kapacitet bygger på p95-ticktiden. Den anden kapacitet kræver i stedet, at serveren i gennemsnit over målevinduet klarer mindst 19,5 af 20 ticks pr. sekund, og at alle bots forbliver forbundet. Begge angiver den mindste værdi af tre gentagelser. En forskel tæller kun, hvis

gentagelsernes intervaller ikke overlapper.

Programmernes kapacitet

Højeste antal bots med i gennemsnit over målevinduet under 50 ms p95-MSPT i alle tre gentagelser, alle med Aikar-flags.



Programmernes kapacitet ved mindst 19,5 TPS

Tilføjet efterfølgende: højeste antal bots med mindst 19,5 TPS i gennemsnit i alle tre gentagelser.



Papers kapacitet pr. JVM-profil

Højeste antal bots med i gennemsnit over målevinduet under 50 ms p95-MSPT i alle tre gentagelser.



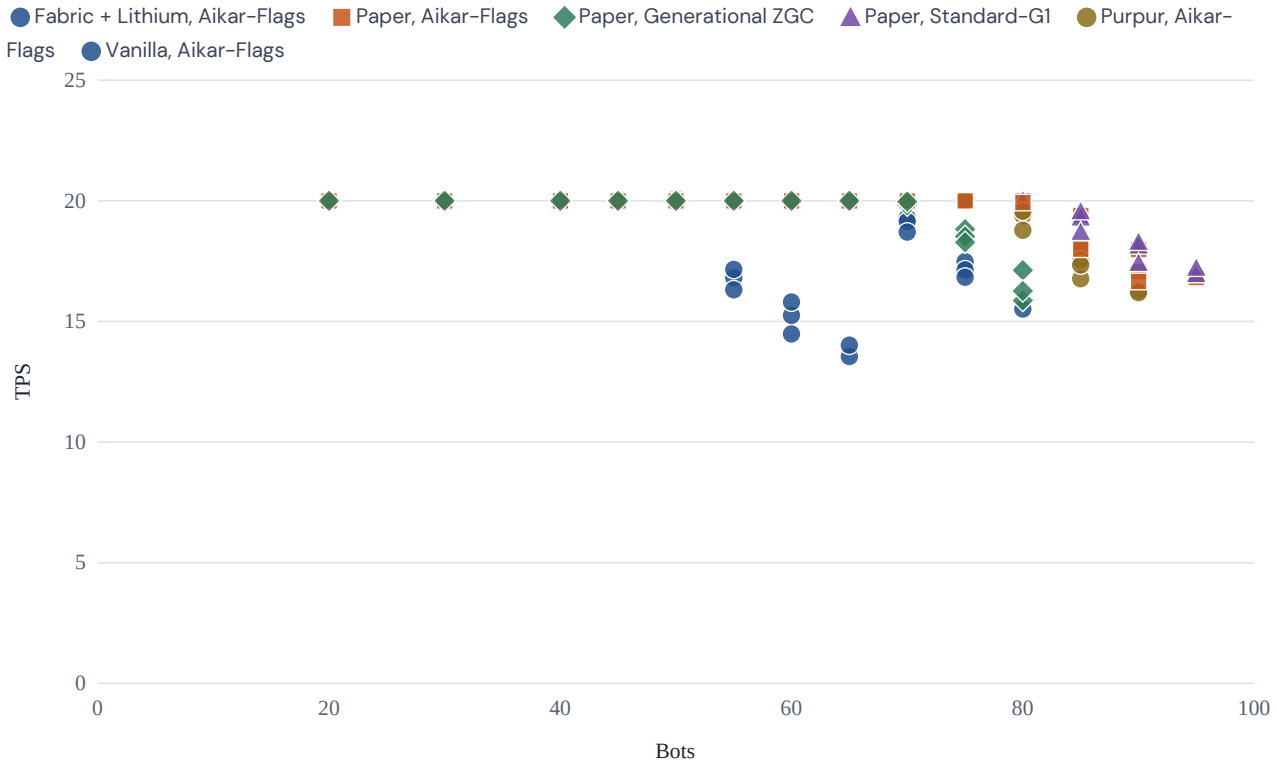
p95-ticktid pr. trin

Hvert punkt er et trin i en kørsel, vist som gennemsnit af målevinduet. Grænsen på 50 ms gælder for dette gennemsnit.



Faktisk taktfrekvens pr. trin

Hvert punkt er et trin i en kørsel. Målet er 20 ticks pr. sekund.



Alle kombinationer

Kombination	Kapacitet	Højeste kørsel	Ved 19,5 TPS	Højeste kørsel (TPS)	p95 ved 50, min	p95 ved 50, maks	p99 ved 50, min	p99 ved 5
Paper, Aikar-Flags	80	85	80	80	27,8 ms	29,8 ms	31,2 ms	
Paper, Standard-G1	80	85	80	85	28,3 ms	29,5 ms	34,4 ms	
Purpur, Aikar-Flags	75	80	75	80	29,4 ms	31,0 ms	33,2 ms	
Paper, Generational ZGC	70	70	70	70	34,1 ms	35,4 ms	39,9 ms	
Fabric + Lithium, Aikar-Flags	65	70	65	65	36,1 ms	37,3 ms	43,4 ms	
Vanilla, Aikar-Flags	55	55	50	50	42,7 ms	43,5 ms	50,2 ms	

Alle 18 kørsler

Nr.	Kombination	Gent.	Kapacitet	TPS der	Ved 19,5 TPS	Gyldig
1	Paper, Aikar-Flags	2	85	19,40	80	Ja
2	Paper, Aikar-Flags	1	80	19,74	80	Ja
3	Paper, Generational ZGC	1	70	20,00	70	Ja
4	Purpur, Aikar-Flags	1	80	19,40	75	Ja
5	Fabric + Lithium, Aikar-Flags	2	70	19,30	65	Ja
6	Purpur, Aikar-Flags	2	75	20,00	75	Ja
7	Paper, Standard-G1	2	85	19,30	80	Ja
8	Vanilla, Aikar-Flags	3	55	16,81	50	Ja
9	Vanilla, Aikar-Flags	1	55	16,31	50	Ja
10	Fabric + Lithium, Aikar-Flags	3	65	20,00	65	Ja
11	Paper, Standard-G1	3	85	19,56	85	Ja
12	Vanilla, Aikar-Flags	2	55	17,16	50	Ja
13	Fabric + Lithium, Aikar-Flags	1	65	20,01	65	Ja
14	Paper, Standard-G1	1	80	20,01	80	Ja
15	Paper, Generational ZGC	3	70	19,82	70	Ja
16	Purpur, Aikar-Flags	3	80	19,57	80	Ja
17	Paper, Aikar-Flags	3	80	19,93	80	Ja
18	Paper, Generational ZGC	2	70	19,97	70	Ja

Parvise sammenligninger

En forskel anses for påvist, hvis intervallerne for de tre gentagelser ikke overlapper.

Nøgletal	A	A min	A maks	B	B min	B maks	Påvist
kapazitaet_p95	Vanilla, Aikar-Flags	55,0	55,0	Paper, Aikar-Flags	80,0	85,0	Ja
kapazitaet_tps	Vanilla, Aikar-Flags	50,0	50,0	Paper, Aikar-Flags	80,0	80,0	Ja
p95_bei_50	Vanilla, Aikar-Flags	42,7	43,5	Paper, Aikar-Flags	27,8	29,8	Ja
p99_bei_50	Vanilla, Aikar-Flags	50,2	55,1	Paper, Aikar-Flags	31,2	33,4	Ja
kapazitaet_p95	Vanilla, Aikar-Flags	55,0	55,0	Purpur, Aikar-Flags	75,0	80,0	Ja
kapazitaet_tps	Vanilla, Aikar-Flags	50,0	50,0	Purpur, Aikar-Flags	75,0	80,0	Ja
p95_bei_50	Vanilla, Aikar-Flags	42,7	43,5	Purpur, Aikar-Flags	29,4	31,0	Ja
p99_bei_50	Vanilla, Aikar-Flags	50,2	55,1	Purpur, Aikar-Flags	33,2	35,0	Ja
kapazitaet_p95	Vanilla, Aikar-Flags	55,0	55,0	Fabric + Lithium, Aikar-Flags	65,0	70,0	Ja
kapazitaet_tps	Vanilla, Aikar-Flags	50,0	50,0	Fabric + Lithium, Aikar-Flags	65,0	65,0	Ja
p95_bei_50	Vanilla, Aikar-Flags	42,7	43,5	Fabric + Lithium, Aikar-Flags	36,1	37,3	Ja
p99_bei_50	Vanilla, Aikar-Flags	50,2	55,1	Fabric + Lithium, Aikar-Flags	43,4	45,0	Ja
kapazitaet_p95	Paper, Aikar-Flags	80,0	85,0	Purpur, Aikar-Flags	75,0	80,0	Nein
kapazitaet_tps	Paper, Aikar-Flags	80,0	80,0	Purpur, Aikar-Flags	75,0	80,0	Nein
p95_bei_50	Paper, Aikar-Flags	27,8	29,8	Purpur, Aikar-Flags	29,4	31,0	Nein
p99_bei_50	Paper, Aikar-Flags	31,2	33,4	Purpur, Aikar-Flags	33,2	35,0	Nein
kapazitaet_p95	Paper, Aikar-Flags	80,0	85,0	Fabric + Lithium, Aikar-Flags	65,0	70,0	Ja
kapazitaet_tps	Paper, Aikar-Flags	80,0	80,0	Fabric + Lithium, Aikar-Flags	65,0	65,0	Ja
p95_bei_50	Paper, Aikar-Flags	27,8	29,8	Fabric + Lithium, Aikar-Flags	36,1	37,3	Ja
p99_bei_50	Paper, Aikar-Flags	31,2	33,4	Fabric + Lithium, Aikar-Flags	43,4	45,0	Ja
kapazitaet_p95	Purpur, Aikar-Flags	75,0	80,0	Fabric + Lithium, Aikar-Flags	65,0	70,0	Ja
kapazitaet_tps	Purpur, Aikar-Flags	75,0	80,0	Fabric + Lithium, Aikar-Flags	65,0	65,0	Ja
p95_bei_50	Purpur, Aikar-Flags	29,4	31,0	Fabric + Lithium, Aikar-Flags	36,1	37,3	Ja
p99_bei_50	Purpur, Aikar-Flags	33,2	35,0	Fabric + Lithium, Aikar-Flags	43,4	45,0	Ja
kapazitaet_p95	Paper, Aikar-Flags	80,0	85,0	Paper, Standard-G1	80,0	85,0	Nein
kapazitaet_tps	Paper, Aikar-Flags	80,0	80,0	Paper, Standard-G1	80,0	85,0	Nein
p95_bei_50	Paper, Aikar-Flags	27,8	29,8	Paper, Standard-G1	28,3	29,5	Nein
p99_bei_50	Paper, Aikar-Flags	31,2	33,4	Paper, Standard-G1	34,4	36,1	Ja
kapazitaet_p95	Paper, Aikar-Flags	80,0	85,0	Paper, Generational ZGC	70,0	70,0	Ja

Nøgletal	A	A min	A maks	B	B min	B maks	Påvist
kapazitaet_tps	Paper, Aikar-Flags	80,0	80,0	Paper, Generational ZGC	70,0	70,0	Ja
p95_bei_50	Paper, Aikar-Flags	27,8	29,8	Paper, Generational ZGC	34,1	35,4	Ja
p99_bei_50	Paper, Aikar-Flags	31,2	33,4	Paper, Generational ZGC	39,9	40,3	Ja
kapazitaet_p95	Paper, Standard-G1	80,0	85,0	Paper, Generational ZGC	70,0	70,0	Ja
kapazitaet_tps	Paper, Standard-G1	80,0	85,0	Paper, Generational ZGC	70,0	70,0	Ja
p95_bei_50	Paper, Standard-G1	28,3	29,5	Paper, Generational ZGC	34,1	35,4	Ja
p99_bei_50	Paper, Standard-G1	34,4	36,1	Paper, Generational ZGC	39,9	40,3	Ja

4 Metode

1 **Fastlæggelse af programmerne**

Alle programmer kørte på Minecraft 26.3. Java, serverprogrammer og Lithium blev kontrolleret med den offentliggjorte kontrolsum. For Fabric-starteren findes der ingen offentliggjort kontrolsum, her er den egne SHA-256-værdi noteret. server.properties var ens for alle programmer, de øvrige indstillinger forblev på det pågældende programs standardværdier. Den eneste yderligere ændring gælder Paper og Purpur: Forbindelsesdrosslen pr. IP-adresse blev slået fra, fordi alle bots kommer fra samme loopback-adresse. Lithium beskriver sig selv som et optimerings-mod, der ikke skal ændre spilmekanikken (kilde 4), Purpur som en direkte erstatning for Paper (kilde 5).

2 **Én fælles verden**

Den uændrede Vanilla-server genererede én gang med seed 20261121 en verden med 31.329 fuldt genererede chunks. Hver kørsel startede fra en frisk kopi af denne verden.

3 **Adskilte kerner**

Serveren kørte fast på kerne 0 til 3, belastningsgeneratoren på kerne 4 til 6. Måle-scriptet var ikke bundet til en kerne. Under målingen blev rådata hvert 15. minut pakket med laveste prioritet på kerne 7. Indtastningen messung: 7 i plan.json blev ikke gennemført som kernebinding. Server og RCON var kun tilgængelige på loopback-adressen, online-mode=false var kun sat i denne isolerede opsætning. Minecraft-EULA'en er godkendt til disse testservere af Erik Friedl.

4 **Protokol-bots**

En egen slank klient taler den rigtige netværksprotokol i version 26.3: login, konfiguration, spilfase, keep-alive og teleport-bekræftelser. Chunk-bekræftelser sender klienten, når den genkender en chunk-batch-meddelelse. Det var tilfældet ved Vanilla og Fabric; ved Paper og Purpur talte den i ingen kørsel en sådan meddelelse. Den protokoldatabase, som mineflayer bruger, havde endnu ikke version 26.3 på måletidspunktet. Hver bot har et fast seed, sender med en måltakt på 20 positionspakker pr. sekund og bevæger sig i gåhastighed (0,2158 blokke pr. takt) mellem tilfældige waypoints. Seks bots danner en gruppe, grupperne ligger 256 blokke fra hinanden. Botsene flyver i kreativ tilstand i højde 120.

5 **Trin og målevindue**

Trinene var 20, 30 og 40 bots, derefter i skridt på fem. Efter tilslutningen fulgte 60 sekunders indsvingningstid og 120 sekunders målevindue med en prøve hvert 5. sekund. En kørsel sluttede efter to trin over grænsen eller under 15 TPS.

6 **Måleparametre**

MSPT p50, p95 og p99 kom for alle programmer fra kommandoen /tick query over RCON, som vurderer de seneste 100 ticks. Trinværdien er gennemsnittet af disse blokke i målevinduet. TPS blev beregnet ud fra spiltidens fremskridt. Dertil kom RAM for Java-processen, CPU pr. kernegruppe, CPU-steal og det via spillerlisten bekræftede antal bots.

7 **Forløb før planen**

Før planen blev fastlagt, kørte to pilotkørsler (Paper, Vanilla) og korte prøvekørsler af alle kombinationer. Derefter blev trinraster og måletider fastlagt, grænsen på 50 ms stod allerede fast på forhånd. Forløbene indgår ikke i analysen.

8 **Gyldighed**

Fastlagt på forhånd: CPU-steal under 2 %, belastningsgenerator under 70 % CPU og under 20 ms timer-forsinkelse (p99), fuldstændige prøver, alle bots forbundet, så længe serveren ikke er overbelastet, og to stikprøve-bots bevæger sig påviseligt i målevinduet. Ugyldige kørsler ville være blevet gentaget, der var ingen.

9 **Kapacitet og forskelle**

Fastlagt på forhånd: Kapaciteten for en kørsel er det højeste trin, op til hvilket alle trin i gennemsnit over målevinduet forbliver under 50 ms p95-MSPT, og alle bots er forbundet. Kapaciteten for en kombination er den mindste værdi af dens tre kørsler. En forskel gælder kun, hvis gentagelsernes intervaller ikke overlapper.

10 Efterfølgende tilføjelse

Efter målingen viste det sig, at taktfrekvensen delvist allerede faldt under 20 TPS, mens p95-ticktiden i gennemsnit stadig lå under 50 ms. Derfor blev kapaciteten ved mindst 19,5 TPS og nøgletallene ved 50 bots yderligere analyseret. Trin 50 er det højeste, hvor alle kombinationer i gennemsnit over målevinduet stadig nåede mindst 19,5 TPS. Disse analyser var ikke fastlagt på forhånd og er markeret som sådan i dataene.

Måleopstilling

Spilversion	Minecraft 26.3 (protokol 777)
Vanilla	26.3
Paper	26.3, build 257
Purpur	26.3, build 2660
Fabric	Loader 0.20.1, Installer 1.1.2
Lithium	0.26.4 til 26.3
Java	Temurin 25.0.5.1+2
Heap	10 GB (Xms lig med Xmx)
JVM-profil standard-G1	-XX:+UseG1GC
JVM-profil Aikar-flags	Flags ifølge PaperMC-dokumentationen
JVM-profil Generational ZGC	-XX:+UseZGC
Virtuel maskine	8 vCPU under KVM, 31,3 GiB RAM, 100 GiB virtuel harddisk, Debian 12
Rapporteret CPU	QEMU Virtual CPU, AuthenticAMD; den fysiske model kan ikke fastslås fra VM'en
Kerner	Server 0 til 3, bots 4 til 6, sikkerhedskopiering 7, måle-script ubundet
Synsafstand	10 chunks, simuleringsafstand 10 chunks
Spiltilstand	Kreativ, sværhedsgrad normal
Måleperiode	21.11.2026, 20:12 UTC til 22.11.2026, 06:59 UTC

5 Stikprøve og periode

Periode	21.11.2026 til 22.11.2026
Studiens population	18 Målekørsler
Evalueret	18 Målekørsler
Udvælgelse	Seks kombinationer af program og JVM-profil med tre gentagelser hver, i en på forhånd fastlagt tilfældig rækkefølge.

6 Begrænsninger i, hvad studien kan vise

Bots er ikke spillere. Botsene flyver, bygger ikke, kæmper ikke, chatter ikke, bruger intet inventar og udløser ingen redstone-kredsløb. Rigtige spillere skaber en anden belastning. De absolutte botantal kan derfor ikke overføres til en rigtig server. Verdenen var genereret på forhånd, og botsene forlod ikke dette område. Belastningen fra generering af nye chunks er derfor ikke med.

Én VM. Der blev målt på en enkelt virtuel maskine med fire kerner til serveren. På anden hardware bliver værdierne anderledes, og afstandene mellem programmerne kan også ændre sig.

Én testprofil. Der var én verden, ét seed, én bevægelsesprofil og bortset fra Lithium ingen mods eller plugins. Programmerne blev ikke optimeret enkeltvis. Med tilpasset konfiguration eller anden belastning kan rækkefølgen se anderledes ud. Dag-nat-skiftet kørte med. Fordi alle kørsler følger samme tidsplan, falder bestemte trin altid i natten. Indflydelsen blev ikke registreret separat i denne måling.

Forskellig standardkonfiguration. Standardkonfigurationen for Paper og Purpur indeholder indstillingen entity-activation-range (dyr og monstre 32 blokke), som mangler i skabelonerne for Vanilla og Fabric. Om alle programmer har beregnet det samme simuleringsarbejde, blev ikke kontrolleret, entitetsantal blev ikke registreret.

MSPT og TPS afviger. Ved Vanilla lå p95-ticktiden ved 55 bots i gennemsnit over målevinduet under 50 ms, men taktfrekvensen på 16,31 til 17,16 TPS. Årsagen blev ikke undersøgt. Ved Vanilla ligger den på forhånd fastlagte kapacitet derfor 5 bots over værdien ved mindst 19,5 TPS. Ved de fem andre kombinationer stemmer begge mindsteværdier overens.

Gennemsnit i stedet for enkeltværdier. Grænsen på 50 ms gælder for gennemsnittet af 100-tick-blokkene i målevinduet. I 13 af 18 kørsler lå i kapacitetstrinnet mindst én enkelt blok på 50 ms eller derover, højst på 54,2 ms. Også grænsen på 19,5 TPS gælder for gennemsnittet af målevinduet. I trinene for kapaciteten ved mindst 19,5 TPS lå enkelte 5-sekunders-prøver under, laveste ved 17,62 TPS (Vanilla, 50 bots).

Drift over målenatten. Ved samme botantal steg p95-ticktiden med positionen i kørselsrækkefølgen, ved 50 bots med omkring 0,11 ms pr. kørsel (analyseret efterfølgende). Årsagen blev ikke undersøgt i denne måling. Rækkefølgen var tilfældig, men ikke afbalanceret i blokke, derfor indeholder gentagelsernes spredning også denne effekt.

Efterfølgende nøgletal. Kapaciteten ved mindst 19,5 TPS, sammenligningstrinnet med 50 bots og p99-sammenligningen mellem Aikar-flags og standard-G1 blev først valgt, efter at dataene var set. De er supplerende observationer og ikke på forhånd fastlagte prøver.

Groft raster, tre gentagelser. Trinene ligger i det relevante område fem bots fra hinanden, mindre forskelle kan ikke måles. Tre gentagelser pr. kombination viser spredningen kun groft. En statistisk

test blev ikke beregnet.

RAM-værdier ikke sammenlignelige. Aikar-flags indeholder AlwaysPreTouch, som reserverer hukommelsen allerede ved start (kilde 1). Den målte arbejdshukommelse afhænger derfor af JVM-indstillingen og viser ikke programmets behov. Denne undersøgelse udtaler sig ikke om, hvilket program der bruger mindre RAM.

JVM-sammenligning kun på Paper. De tre JVM-profiler blev kun målt med Paper, en heap på 10 GB og Java 25. Programsammenligningen kørte kun med Aikar-flags. Begge resultater kan ikke uden videre overføres til andre kombinationer.

Øjebliksbillede. Resultaterne gælder for de nævnte versioner og builds. Senere versioner kan opføre sig anderledes.

7 Data og kontrollerbarhed

For at gentage målingen skal man bruge en Linux-maskine med otte kerner og omkring 32 GB RAM som her, Node.js og Python 3. Scripts forventer arbejdsmappen /root/studien/haertetest, henter Java og de samme programversioner, så længe disse downloads er tilgængelige, kontrollerer kontrolsummerne, genererer verdenen og afvikler måleplanen. Seed, rækkefølge, trin og regler står i plan.json. Anden hardware, andre builds eller en anden belastning af værtssystemet giver andre absolutte værdier.

Fil	Størrelse	SHA-256
daten.csv Måleværdier pr. trin (CSV)	35,6 KB	a2c2bfdfcc8ddfce2f6eac4044c15aeb ed57f4d0fab95b8a48af72c47e2f9e9e
daten.json Måleværdier pr. trin (JSON)	94,6 KB	f798d400aba35617cf30f8c0cd2ad07c f6aace6e19e02300a0a1b822c6ac980e
laeufe.csv Resultat pr. kørsel	2,7 KB	3a7f0af91987e982f8e95f97c60ef833 0ea9ba8a9f4688e76c64f8514b3486d1
zusammenfassung.csv Resultat pr. kombination	1,1 KB	8ba8269c39e20d767c2a8c87dfb805fd 0d937bb59bd448680f23401f0e66bd88
vergleiche.csv Parvise sammenligninger	4,7 KB	99575f03f6a3f2075bcd20426a18e86d 6d14ad5410618ebb101050c3fabcda9d
p99-aikar-g1.csv p99 Aikar-flags og standard-G1	0,5 KB	4d0b6c017068a81642cddc339b1623f2 71d5d543abaf0ea9e4722152907e64ac
proben.csv Enkeltprøver	517,9 KB	7fa08cf87e94c1e58c680ae9cec668ba 6d56da228bda6be08849d8232db27c33
plan.json Måleplan	5,9 KB	cc335b804f52374f6c77afda5c55dd19 dbf5fa03d595c8f3a9bb72cedd569192

Fil	Størrelse	SHA-256
messprotokoll.txt Målelog	8,1 KB	f234fa1d0a16b312777dd0b93fb12f0e b13161fbdbb13125501e034837705a64
herkunft.txt Programmernes oprindelse	1,4 KB	5f9e921b1c4fe697d2e676b7b5777eee ae784a4c4418b211230e7ac333bb676d
liesmich.txt Læsmig	1,4 KB	8a444fae54933207d6f896459ac23511 c541a656d0fc2da2a26b2f69091f0935
01-einrichten.sh Opsætning	3,4 KB	27627241f4e7964be26e9b3ac2b3c1de 74482e323348569f6e7ba6e9290b7ea9
02-vorlagen.py Verden og skabeloner	4,1 KB	fc70d06d68b18d70425e0987794f06da 257659424a168433c2048577258d4cd5
03-plan.py Måleplan	3,5 KB	773d40d0d4dd1323c748a8c383ecf1ff b0fb91299cbea545f21144dfa083a034
gemeinsam.py Fælles dele	5,3 KB	da16b51f5b78fd55c070c1eaa9b43485 cd7fd1654ce8c47e71df87b26bcb7e20
messlauf.py Målekørsel	10,9 KB	395ec326a334855d03a333879da16f90 0730d0307f06065f078f79af8de21637
hauptlauf.py Hovedmåling	4,6 KB	7b1602088aba30734e706e35ab894d4c 918a91eec8b0846a68a2ee392c6cec2f
bots.mjs Belastningsgenerator	14,3 KB	6fa44c58b3c56a31fb77fb19602cc06a ab93f08e8e8984a40bb95f2ad1064a10
welt-pruefen.py Verdenskontrol	1,5 KB	e0bb7dcc1e2327e9b916355b3e3dba55 2751a2f6c13989ffd07f10c234c0940e
auswertung.mjs Analyse	10,0 KB	42dc3452a2061bad7ede2d84610cdf8c 8ef3adb5e7c6c7b8593c502aa2b85a33
baue-studie.mjs Undersøgelsesbygning	39,2 KB	c0e76feab583afa11baf3bbce105203f 4a3ebcf57e66adf44380bb7854a8b90b

Datafilernes felter

Felt	Enhed	Betydning
lauf		Betegnelse for målekørslen med position i den tilfældige rækkefølge
kombination		Betegnelse bestående af program og JVM-profil
anzeige		Læsbart navn på kombinationen
programm		Serverprogram
jvm		JVM-profil
wiederholung		Nummer på gentagelsen, 1 til 3
versuch		Forsøg inden for gentagelsen, 1 er det første

Felt	Enhed	Betydning
reihenfolge		Position i den fastlagte tilfældige rækkefølge
gueltig		Kørslen opfylder alle på forhånd fastlagte gyldighedsregler
lauf_gueltig		Den tilhørende kørsel er gyldig
spieler	Bots	Målantal bots i dette trin
spieler_min	Bots	Mindste via spillerlisten bekræftede antal bots i målevinduet
spieler_verbunden	Bots	Via spillerlisten bekræftet antal bots i denne prøve
unter_grenze		p95-MSPT i gennemsnit over målevinduet under 50 ms og alle bots forbundet
p50_ms	ms	Median af ticktiden; i trin gennemsnit af 100-tick-blokkene
p95_ms	ms	95. percentil af ticktiden; i trin gennemsnit af 100-tick-blokkene
p95_max_ms	ms	Højeste p95-værdi for en 100-tick-blok i målevinduet
p99_ms	ms	99. percentil af ticktiden; i trin gennemsnit af 100-tick-blokkene
mspt_mittel_ms	ms	Gennemsnitlig ticktid ifølge /tick query
tps	Ticks/s	Faktisk taktfrekvens ud fra spiltidens fremskridt; i trin gennemsnit af målevinduet
tps_min	Ticks/s	Laveste taktfrekvens for en 5-sekunders-prøve i målevinduet
ram_gb	GiB	Optaget arbejdshukommelse for Java-processen (RSS)
cpu_server_pct	%	Gennemsnitlig udnyttelse af de fire serverkerner
cpu_bots_pct	%	Gennemsnitlig udnyttelse af belastningsgeneratorens tre kerner
steal_pct	%	CPU-steal over alle otte kerner
bots_verzug_p99_ms	ms	Højeste p99-forsinkelse af timerne i belastningsgeneratoren
getrennt	Bots	Uventet frakoblede bots op til dette trin
proben		Antal 5-sekunders-prøver i målevinduet
zeit_unix	s	Tidspunkt for prøven som Unix-tid
erste_ueberlast	Bots	Første trin, hvis gennemsnitlige p95-MSPT ligger over grænsen på 50 ms

Felt	Enhed	Betydning
kapazitaet	Bots	På forhånd fastlagt kapacitet; i sammenfatningen mindste værdi af de tre kørsler
tps_bei_kapazitaet	Ticks/s	Gennemsnitlig taktfrekvens i kapacitetstrinnet
kapazitaet_tps	Bots	Højeste trin, op til hvilket alle trin i gennemsnit over målevinduet holdt mindst 19,5 TPS; i sammenfatningen mindste værdi
kapazitaet_tps_190	Bots	Som kapazitaet_tps, med 19,0 TPS som grænse
kapazitaet_tps_198	Bots	Som kapazitaet_tps, med 19,8 TPS som grænse
p95_bei_50_ms	ms	p95-MSPT i trinnet med 50 bots
p99_bei_50_ms	ms	p99-MSPT i trinnet med 50 bots
tps_bei_50	Ticks/s	Taktfrekvens i trinnet med 50 bots
stufen_gemessen		Antal målte trin i kørslen
ungueltig_grund		Årsag ved ugyldige kørsler, ellers tom
im_programmvergleich		Kombinationen hører til sammenligningen af programmerne
im_jvm_vergleich		Kombinationen hører til sammenligningen af JVM-profilerne
laeufe_gueltig		Gyldige kørsler for kombinationen
laeufe_ungueltig		Ugyldige kørsler for kombinationen
kapazitaet_median	Bots	Median af kapaciteten over kørslerne
kapazitaet_max	Bots	Største kapacitetsværdi for kørslerne
kapazitaet_tps_median	Bots	Median af kapazitaet_tps
kapazitaet_tps_max	Bots	Største værdi af kapazitaet_tps
tps_bei_kapazitaet_min	Ticks/s	Laveste taktfrekvens ved kapaciteten over kørslerne
p95_bei_50_min_ms	ms	Mindste p95-værdi ved 50 bots
p95_bei_50_max_ms	ms	Største p95-værdi ved 50 bots
p99_bei_50_min_ms	ms	Mindste p99-værdi ved 50 bots
p99_bei_50_max_ms	ms	Største p99-værdi ved 50 bots
ram_bei_50_gb	GiB	Gennemsnitlig RSS ved 50 bots
cpu_bei_50_pct	%	Gennemsnitlig udnyttelse af serverkernerne ved 50 bots
vergleich		Programsammenligning eller JVM-sammenligning
kennzahl		Sammenlignet nøgletal

Felt	Enhed	Betydning
regel		Om nøgletallet blev fastlagt på forhånd eller tilføjet efterfølgende
a		Første kombination
a_min		Mindste værdi for den første kombination
a_max		Største værdi for den første kombination
b		Anden kombination
b_min		Mindste værdi for den anden kombination
b_max		Største værdi for den anden kombination
unterschied_belegt		Gentagelsernes intervaller overlapper ikke
besser		Kombinationen med den gunstigere værdi, kun ved påvist forskel
aikar_min	ms	Mindste p99-værdi med Aikar-flags i trinnet
aikar_max	ms	Største p99-værdi med Aikar-flags i trinnet
g1_min	ms	Mindste p99-værdi med standard-G1 i trinnet
g1_max	ms	Største p99-værdi med standard-G1 i trinnet
aikar_niedriger		Alle tre Aikar-værdier ligger under alle tre G1-værdier
g1_niedriger		Alle tre G1-værdier ligger under alle tre Aikar-værdier

8 Kilder

[1] PaperMC (2026): Aikar's flags.
https://docs.papermc.io/paper/aikars-flags/
Hentet den 21.11.2026

[2] Minecraft Wiki (2026): Commands/tick.
https://minecraft.wiki/w/Commands/tick
Hentet den 21.11.2026

[3] OpenJDK: JEP 490: ZGC: Remove the Non-Generational Mode.
https://openjdk.org/jeps/490
Hentet den 21.11.2026

[4] Lithium-Projekt auf Modrinth (2026): Lithium.
https://modrinth.com/mod/lithium
Hentet den 21.11.2026

[5] PurpurMC: Purpur Documentation.
https://purpurmc.org/docs/purpur/
Hentet den 21.11.2026

9 Åbenhed

Egen interesse

Nexthosting sælger selv gameservere, herunder servere til Minecraft, og har derfor en økonomisk interesse i dette emne. Ingen af de testede programmer stammer fra Nexthosting. Målingen kørte på en test-VM, som Nexthosting har bestilt. Undersøgelsen vurderer ikke hosting-udbyderes tilbud. Ansvarlig for denne undersøgelse er Nexthosting, kontakt: support@nexthosting.net.

Forfatterskab

Erik Friedl, Nexthosting. Udgiver er Nexthosting.

Licens

CC BY 4.0: https://creativecommons.org/licenses/by/4.0/deed.da

10 Sådan citerer du denne studie

APA

Friedl, E. & Nexthosting. (2026). Serverbelastningstest: Hvor mange protokol-bots kan Vanilla, Paper, Purpur og Fabric med Lithium bære? (Version 1.0.0) [Rapport]. Nexthosting. https://nexthosting.net/da/undersoegelser/server-haertetest-minecraft

BibTeX

```
@techreport{nexthosting2026serverhaertetestminecraft,
  author     = {Erik Friedl and {Nexthosting}},
  title      = {Serverbelastningstest: Hvor mange protokol-bots kan Vanilla, Paper, Purpur og Fabric med
Lithium bære?},
  institution= {Nexthosting},
  type       = {Studie},
  year       = {2026},
  version    = {1.0.0},
  url        = {https://nexthosting.net/da/undersoegelser/server-haertetest-minecraft},
  urldate    = {2026-11-22},
  language   = {danish}
}
```

11 Ændringer og rettelser

Version	Dato	Type	Ændring
1.0.0	22.11.2026	Version	Første udgave.