

# iH ACC

技术白皮书

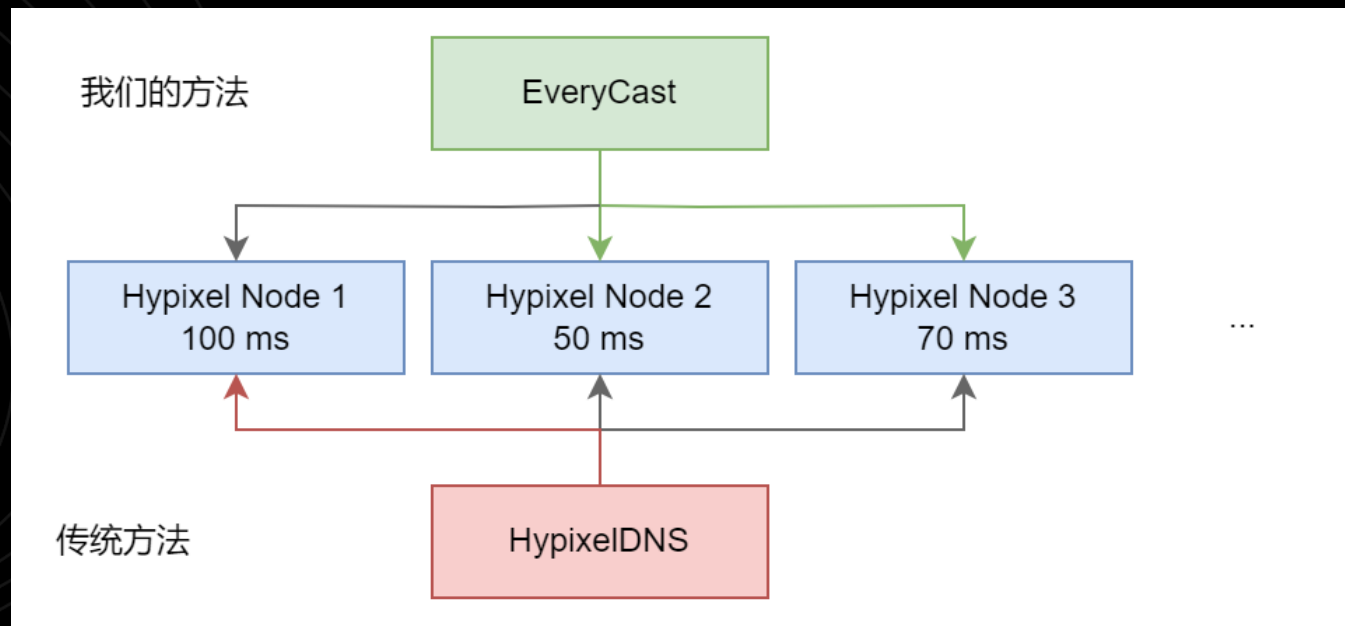
## Feat. HeaderMod (L4-5)



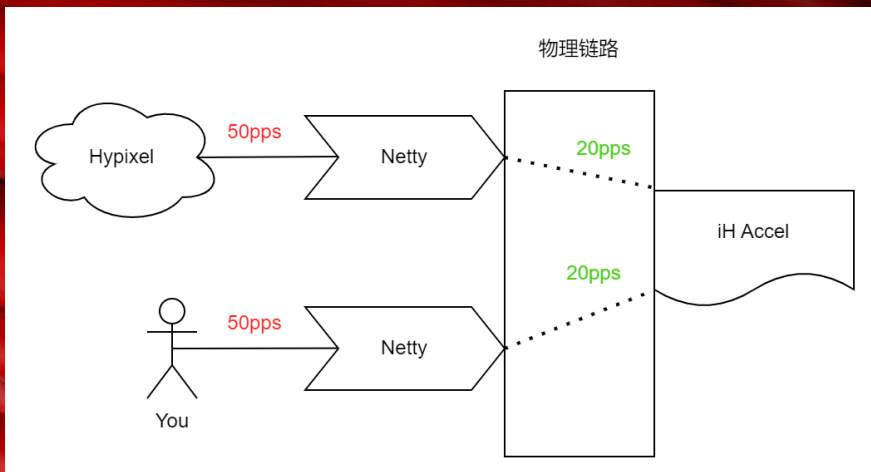
- Hypixel服务器会严格校验包头中的域名信息以确保连接合法性，直接使用Nginx无法转发。
- 我们的解决方案在网络层对数据包头部进行了智能替换，确保域名校验能够顺利通过，同时完全不影响正版登录流程 and 安全性。
- 整个过程对终端用户完全透明，玩家无需进行任何额外配置即可享受更快的连接速度。
- 亦实现了在首次登录时展现自定义公告的功能。

## Feat. EveryCast (L5)

- 彻底重写了Hypixel基于DNS的AnyCast网络连接逻辑。
- 直接连接到每一个Hypixel源站节点。
- 开发了实时测速系统，持续监控所有可用解析IP的延迟、丢包率和连接质量，根据玩家实际网络环境动态选择响应最快的服务器节点。
- 采用智能负载均衡算法在多条优质线路间分配流量，在网络高峰期能有效避开拥堵节点。
- 大幅减少连接中断情况，为玩家提供更加流畅稳定的游戏体验。








## Feat. ANaggle (L4)

- 基于标准Naggle算法的粘包，对Netty透明，无需客户端修改。
- 通过精细控制数据包发送时机和大小，在保持游戏响应速度的同时减少了30%以上的网络开销。
- 降低TCP PPS，大幅优化丢包问题。

# Feat. MCQoS (L4-5)



## 世界加载场景

- 增大缓存区大小
- 减小窗口大小
- 减慢世界加载速度，避免流量尖刺带来的掉线问题与对其他用户的影响
- 提升带宽利用率，最终加速世界加载

## PVP场景

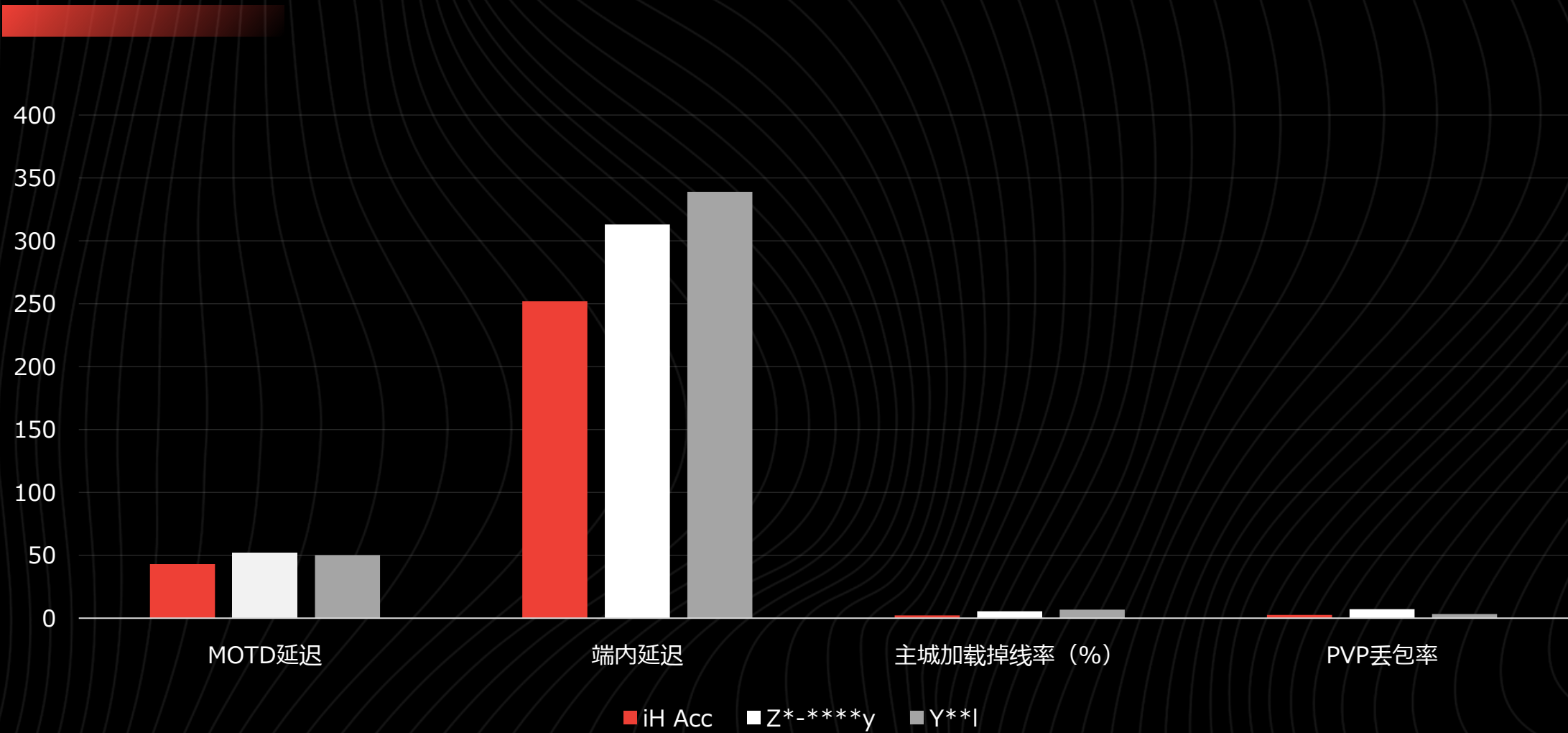
- 减小缓存区大小
- 增大窗口大小
- 降低延迟与抖动
- 优化实时性，避免吞刀、卡方块等高血压问题

## Feat. NeverSleep (L4)

智能TCP空包管理机制，动态调整保持连接活跃的空包发送策略。当检测到游戏业务数据包较少时，系统会自动增加空包发送频率，防止连接被运营商误判为低优先级长连接而限速。

随网络状况动态调整空包间隔，在保证连接活跃的前提下尽可能节省流量。

# 与其他方案的对比



# 源代码不开放说明



- 核心优化技术采用Go语言实现，大量使用了unsafe指针操作和系统级syscall调用，代码可读性较低，即使对经验丰富的Go开发者而言也难以安全地理解和修改，还容易引入安全隐患因此暂不考虑开源原始代码。
- 作为替代方案，我们将定期发布详细的技术白皮书，完整说明各优化模块的工作原理和实现思路，同时提供经过严格测试的release版本供用户直接使用。



# 致谢

- 星遥游戏



- Team iH



- Head Dev: huzpsb