

DNS 缓存系统软件需求规格说明书

2011-6-24

目 录

1	概述	2
1.1	编写目的	2
1.2	系统目标	2
2	任务概述	2
2.1	目标	2
2.2	开发时限	2
3	系统需求说明	3
3.1	系统边界	3
3.2	功能需求	3
3.3	性能需求	4
3.4	实现要求	4
3.5	安全性	4
4	功能划分	4
4.1	内存管理模块	4
4.2	缓存管理模块	5
4.3	线程池管理及线程调度模块	5
4.4	通信协议	5
4.5	安全模块	5
4.6	并发访问和同步机制	5
4.7	客户端	5
4.8	自动化测试	5
5	验收标准	5
5.1	验收标准	5

1 概述

1.1 编写目的

本文档是李琳小组按 DNS 缓存项目需求的编制的。本文档的编写为下阶段的设计、开发提供依据，为项目组成员对需求的详尽理解，以及在开发开发过程中的协同工作提供强有力的保证。同时本文档也作为项目评审验收的依据之一。

1.2 系统目标

场景描述：在百度的大规模集群环境中，有这样一个模块，在它工作时，需要根据用户指定的机器名列表，与这些机器建立连接完成一定任务。在这个过程中，会大量进行域名的解析，不仅对 DNS 服务器造成较大压力，还容易发生解析失败的情况，设计一个系统，来应对突发性的、海量的 DNS 请求并在超出能力极限的情况下自我保护。该系统同时可以应用于短连接大并发的 DNS 请求并在超出能力极限的情况下自我保护。

2 任务概述

2.1 目标

本系统为其它系统服务，根据客户提出的域名，返回域名对应的 IP 地址。用户可以一次提出一个请求，也可以在一个请求中请求若干域名的 IP 地址。系统根据用户的请求，从 DNS 服务器获得域名和 IP 地址的对应关系，并进行缓存，从而在用户下次请求时，可以快速响应。当服务器的缓存放满之后，就使用 LIRS 算法，对缓冲区进行更新。

2.2 开发时限

开发周期为 25 天，最后提交期限为 2011 年 7 月 12 号。

3 系统需求说明

3.1 系统边界

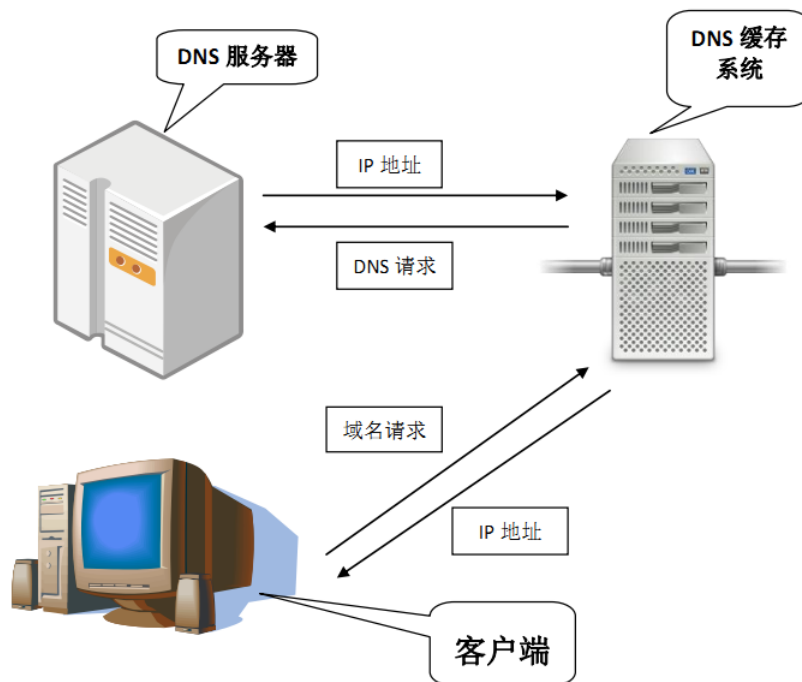


图 1：系统边界

3.2 功能需求

1. 可配置的缓存更新策略，如缓存的老化周期，缓存的大小。根据 LIRS 来实现缓存更新算法，并可扩展。
2. 处理网络请求，请求内容为一个或者 n 个主机名，返回对应的 ip 地址。包格式请考虑以后的扩展性。
3. 利用多核 CPU，采用多线程实现。
4. 输入/输出：客户端的请求作为输入（其中包含请求所域名的计数以及每个请求域名的长度、请求内容），系统根据是否能在缓存中直接

命中将输出分为两部分：可快速响应并返回给客户端的输出，需要请求 DNS 服务器并返回的输出。

3.3 性能需求

1. 在 8*2.33G CPU，16G 内存，64 位主机上处理速度需要达到 5000req/s。此数值将根据具体主机的不同而调整，基本评判标准是在相同缓存策略下，处理速度越快越好。
2. 对于超出系统承受能力的请求，使用拒绝策略来减少对正常服务的提供的影响。

3.4 实现要求

1. 文档齐全：设计文档、单元测试文档、自动测试文档。文档的撰写思路明晰，内容完善。
2. 设计方面考虑完整，模块划分清晰。
3. 使用 C 实现。实现功能完整，代码符合代码规范，好的代码的可读性和可维护性，必要的注释。
4. 执行单元测试。
5. 完善的自动化测试用例。撰写程序对整体功能进行测试。

3.5 安全性

保证服务器数据安全，不受 DDoS 攻击。

4 功能划分

4.1 内存管理模块

给返回报文预分配内存，因此需要内存管理模块，进行预分配内存的管理，碎片整理，和防止内存泄漏。。

4.2 缓存管理模块

本系统将从 DNS 服务器获取的域名和 IP 地址的对应关系缓存在内存中，采用 LIRS 算法驱动的缓存管理，提高缓存命中率。

4.3 线程池管理及线程调度模块

系统在启动时，预先分配固定数目的线程，放入线程池。当有客户连接请求域名时，从线程池中申请空闲线程，线程工作完毕后将线程放回线程池中。

4.4 通信协议

指定客户端与系统的通信协议并使用协议通信。

4.5 安全模块

预防 DDoS 攻击。

4.6 并发访问和同步机制

线程间数据并发访问处理，和线程间的同步机制。

4.7 客户端

编写测试客户端，遵从通信协议与系统交互，客户端发送 DNS 解析请求并接收和处理服务器的返回报文。

4.8 自动化测试

编写自动化测试客户端，测试系统的功能性需求和非功能性需求。

5 验收标准

5.1 验收标准

- 实现所有功能需求

- 满足非功能性需求
- 系统设计文档完整，且符合规范
- 代码符合规范，且与系统设计一致