



(12) 发明专利申请

(10) 申请公布号 CN 104219215 A

(43) 申请公布日 2014. 12. 17

(21) 申请号 201310220255. 4

(22) 申请日 2013. 06. 05

(71) 申请人 深圳市腾讯计算机系统有限公司

地址 518057 广东省深圳市南山区高新区高
新南一路飞亚达大厦 5-10 楼

(72) 发明人 余子军 刘贤彬

(74) 专利代理机构 上海波拓知识产权代理有限
公司 31264

代理人 杨波

(51) Int. Cl.

H04L 29/06(2006. 01)

H04L 1/16(2006. 01)

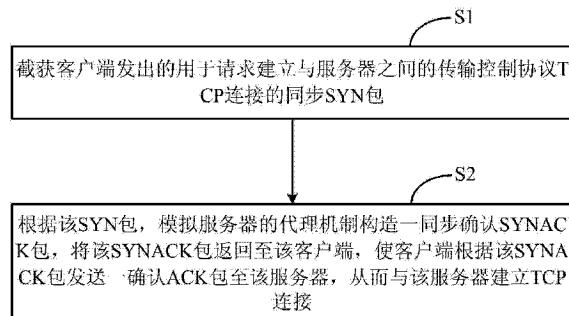
权利要求书2页 说明书5页 附图5页

(54) 发明名称

一种 TCP 连接的建立方法、装置、终端、服务
器及系统

(57) 摘要

本发明涉及一种 TCP 连接的建立方法，包括：截获该客户端发出的用于请求建立与服务器之间的传输控制协议 TCP 连接的同步 SYN 包；根据该 SYN 包，模拟服务器的代理机制构造一同步确认 SYNACK 包，将该 SYNACK 包返回至该客户端，使客户端根据该 SYNACK 包发送一确认 ACK 包至该服务器，从而与该服务器建立 TCP 连接。本发明还提供一种 TCP 连接的建立装置、终端、服务器及系统。利用本发明可以减少 TCP 连接建立过程中客户端与服务器之间握手交互的次数，提高建立 TCP 连接的速度，并降低服务器的内存消耗。



1. 一种 TCP 连接的建立方法,其特征在于,该方法包括以下步骤:

截获客户端发出的用于请求建立与服务器之间的传输控制协议 TCP 连接的同步 SYN 包;

根据该 SYN 包,模拟服务器的代理机制构造一同步确认 SYNACK 包,将该 SYNACK 包返回至该客户端,使客户端根据该 SYNACK 包发送一确认 ACK 包至该服务器,从而与该服务器建立 TCP 连接。

2. 如权利要求 1 所述的 TCP 连接的建立方法,其特征在于,所述服务器的代理机制为 syn-cookies 机制。

3. 如权利要求 1 所述的 TCP 连接的建立方法,其特征在于,所述 SYN 包携带有一序列号 x。

4. 如权利要求 3 所述的 TCP 连接的建立方法,其特征在于,所述模拟服务器的代理机制构造一同步确认 SYNACK 包的步骤包括:

通过对该序列号 x 加 1 生成一确认号 y,并使用预定算法生成一序列号 a,然后生成携带该确认号 y 和序列号 a 的该 SYNACK 包。

5. 如权利要求 4 所述的 TCP 连接的建立方法,其特征在于,所述客户端接收到该 SYNACK 包后,将先根据该确认号 y 验证该 SYNACK 包是否合法,若该 SYNACK 包合法,再向服务器发送该 ACK 包。

6. 如权利要求 5 所述的 TCP 连接的建立方法,其特征在于,该 ACK 包携带有一确认号 q。

7. 如权利要求 6 所述的 TCP 连接的建立方法,其特征在于,所述服务器接收到该 ACK 包后,将根据该确认号 q 通过所述代理机制验证该 ACK 包是否合法,若该 ACK 包合法,则与该客户端建立 TCP 连接。

8. 如权利要求 7 所述的 TCP 连接的建立方法,其特征在于,所述服务器验证该 ACK 包是否合法的步骤包括:

通过所述代理机制使用所述的预定算法生成一个数值;

判断该确认号 q 减 1 后与该数值是否相等,若是,则验证该 ACK 包合法。

9. 如权利要求 1 所述的 TCP 连接的建立方法,其特征在于,所述客户端将该 ACK 包与待发送数据共同发送至服务器。

10. 一种 TCP 连接的建立装置,其特征在于,该装置包括:

截获模块,用于截获客户端发出的用于请求建立与服务器之间的传输控制协议 TCP 连接的同步 SYN 包;

构造模块,用于根据该 SYN 包,模拟服务器的代理机制构造一同步确认 SYNACK 包,将该 SYNACK 包返回至该客户端,使客户端根据该 SYNACK 包发送一确认 ACK 包至该服务器,从而与该服务器建立 TCP 连接。

11. 如权利要求 10 所述的 TCP 连接的建立装置,其特征在于,所述服务器的代理机制为 syn-cookies 机制。

12. 如权利要求 10 所述的 TCP 连接的建立装置,其特征在于,所述 SYN 包携带有一序列号 x。

13. 如权利要求 12 所述的 TCP 连接的建立装置,其特征在于,所述模拟服务器的代理机

制构造一同步确认 SYNACK 包包括：

通过对该序列号 x 加 1 生成一确认号 y，并使用预定算法生成一序列号 a，然后生成携带该确认号 y 和序列号 a 的该 SYNACK 包。

14. 如权利要求 13 所述的 TCP 连接的建立装置，其特征在于，所述客户端接收到该 SYNACK 包后，将先根据该确认号 y 验证该 SYNACK 包是否合法，若该 SYNACK 包合法，再向服务器发送该 ACK 包。

15. 如权利要求 14 所述的 TCP 连接的建立装置，其特征在于，该 ACK 包携带有一确认号 q。

16. 如权利要求 15 所述的 TCP 连接的建立装置，其特征在于，所述服务器接收到该 ACK 包后，将根据该确认号 q 通过所述代理机制验证该 ACK 包是否合法，若该 ACK 包合法，则与该客户端建立 TCP 连接。

17. 如权利要求 16 所述的 TCP 连接的建立装置，其特征在于，所述服务器验证该 ACK 包是否合法包括：

通过所述代理机制使用所述的预定算法生成一个数值；

判断该确认号 q 减 1 后与该数值是否相等，若是，则验证该 ACK 包合法。

18. 如权利要求 10 所述的 TCP 连接的建立装置，其特征在于，所述客户端将该 ACK 包与待发送数据共同发送至服务器。

19. 一种终端，其特征在于，所述终端包括如权利要求 10-18 任一项所述的 TCP 连接的建立装置。

20. 一种系统，其特征在于，所述系统包括至少一个服务器，和至少一个如权利要求 19 所述的终端。

21. 一种服务器，其特征在于，所述服务器包括如权利要求 10-18 任一项所述的 TCP 连接的建立装置。

22. 一种系统，其特征在于，所述系统包括至少一个终端，至少一个服务器，所述至少一个终端和所述至少一个服务器通过至少一个如权利要求 21 所述的服务器连接。

一种 TCP 连接的建立方法、装置、终端、服务器及系统

技术领域

[0001] 本发明具体实施例涉及网络通信技术领域，特别涉及一种传输控制协议 TCP 连接的建立方法、装置、终端、服务器及系统。

背景技术

[0002] 如图 1 所示，传输控制协议 (Transmission Control Protocol, TCP) 规定的 TCP 连接建立过程的三次握手 (Three-way Handshake) 包括：

[0003] 第一步，客户端发送一个请求建立 TCP 连接的同步 (Synchronize, SYN) 包给服务器，该 SYN 包中 TCP 首部的 SYN 标志位被置 1，并且该 SYN 包携带有 1 个序列号 (sequence number) x；

[0004] 第二步，服务器在接收到该 SYN 包后，将响应一个同步确认 (SYNACK) 包给客户端，表示客户端的请求被接受，该 SYNACK 包中 TCP 首部的 SYN 标志位和 ACK 标志位都被置 1，并且该 SYNACK 包中携带有 1 个确认号 (acknowledge number) y 以及 1 个序列号 z；

[0005] 第三步，客户端接收到该 SYNACK 包后，检查该 SYNACK 包中的确认号 y 是否等于 x+1，若是，则发送一个确认 (Acknowledge, ACK) 包给服务器，该 ACK 包中 TCP 首部的 ACK 标志位被置 1，且该 ACK 包携带有 1 个确认号 q。该 ACK 包还可以附带待发送的数据，例如基于 HTTP (Hypertext transfer protocol, 超文本传输协议) 的请求 (Request) 数据。服务器接收到该 ACK 包后，若检查到该确认号 q 等于 z+1，则 TCP 连接建立成功，服务器接收并响应该数据。

[0006] 上述的三次握手的过程需要客户端与服务器进行多次交互，经过一定的往返时间 (Round-trip Time, 简称 RTT) 的延迟才能成功建立 TCP 连接。因此，建立 TCP 连接的速度将受到 RTT 延迟的影响。若 TCP 连接建立的速度较迟缓，会使服务器对客户端发送数据的响应也较迟缓。客户端需要等待一个 RTT 延迟才能在该 ACK 包中附带数据发送至服务器，并且再等待一个 RTT 延迟才能接收到服务器对该数据的响应。

[0007] 为了减少 RTT 延迟，加速 TCP 连接的建立，Linux 系统下的 TCP Fast Open (简称 TFO) 选项支持在服务器与客户端通过三次握手已经建立过一次 TCP 连接的情况下，若该 TFO 选项被开启，则当服务器再次接收到客户端发送的要求与服务器建立 TCP 连接的 SYN 包时，若验证该客户端已经与服务器成功建立过 TCP 连接，则可以直接建立与客户端的 TCP 连接，以及接收该 SYN 包附带的数据并响应。然而，该 TFO 选项还是要在客户端与服务器通过三次握手的交互过程预先建立过 TCP 连接的基础上才能应用。

[0008] 此外，上述的三次握手过程需要在服务器中开辟一个较大的内存空间用于存储服务器在每接收到一个 SYN 包后产生的半连接状态的条目。随着服务器接收到 SYN 包的数量不断增加，将会大量消耗服务器的内存资源。

发明内容

[0009] 有鉴于此，有必要提供一种 TCP 连接的建立方法、装置、终端、服务器及系统，可以

减少 TCP 连接建立过程中客户端与服务器之间握手交互的次数, 提高建立 TCP 连接的速度, 并降低服务器的内存消耗。

[0010] 一种 TCP 连接的建立方法, 包括以下步骤: 截获客户端发出的用于请求建立与服务器之间的传输控制协议 TCP 连接的同步 SYN 包; 根据该 SYN 包, 模拟服务器的代理机制构造一同步确认 SYNACK 包, 将该 SYNACK 包返回至该客户端, 使客户端根据该 SYNACK 包发送一确认 ACK 包至该服务器, 从而与该服务器建立 TCP 连接。

[0011] 一种 TCP 连接的建立装置, 包括: 截获模块, 用于截获客户端发出的用于请求建立与服务器之间的传输控制协议 TCP 连接的同步 SYN 包; 构造模块, 用于根据该 SYN 包, 模拟服务器的代理机制构造一同步确认 SYNACK 包, 将该 SYNACK 包返回至该客户端, 使客户端根据该 SYNACK 包发送一确认 ACK 包至该服务器, 从而与该服务器建立 TCP 连接。

[0012] 一种终端, 包括如上所述的 TCP 连接的建立装置。一种系统, 包括至少一个服务器和至少一个如前述的终端。

[0013] 一种服务器, 包括如上所述的 TCP 连接的建立装置。一种系统, 包括至少一个终端, 至少一个服务器, 所述至少一个终端和所述至少一个服务器通过至少一个如前述的服务器连接。

[0014] 相较于现有技术, 本发明 TCP 连接的建立方法、装置、终端、服务器及系统, 通过截获客户端发送的 SYN 包, 并模拟服务器向客户端返回 SYNACK 包, 从而建立客户端与服务器之间的 TCP 连接, 可以减少 TCP 连接建立过程中客户端与服务器之间握手交互的次数, 提高建立 TCP 连接的速度, 并降低服务器的内存消耗。

[0015] 为让本发明的上述和其他目的、特征和优点能更明显易懂, 下文特举较佳实施例, 并配合所附图式, 作详细说明如下。

附图说明

[0016] 图 1 为建立 TCP 连接的三次握手的过程示意图。

[0017] 图 2 为本发明第一实施例提供的 TCP 连接的建立方法应用时的环境示意图。

[0018] 图 3 为本发明第一实施例提供的 TCP 连接的建立方法的流程图。

[0019] 图 4 为本发明第二实施例提供的 TCP 连接的建立装置的框图。

[0020] 图 5 为客户端与服务器通过图 4 中的 TCP 连接的建立装置建立 TCP 连接的交互过程示意图。

[0021] 图 6 为本发明第三实施例提供的一种终端的框图。

[0022] 图 7 为本发明第四实施例提供的一种系统的框图。

[0023] 图 8 为本发明第五实施例提供的一种服务器的框图。

[0024] 图 9 为本发明第六实施例提供的一种系统的框图。

具体实施方式

[0025] 为更进一步阐述本发明为实现预定发明目的所采取的技术手段及功效, 以下结合附图及较佳实施例, 对依据本发明的具体实施方式、结构、特征及其功效, 详细说明如后。

[0026] 参阅图 2 所示, 为本发明第一实施例提供的 TCP (Transmission Control Protocol, 传输控制协议) 连接的建立方法应用时的环境示意图。在本实施例中, 该 TCP 连

接的建立方法用于在服务器 2 与终端 1 中的客户端 10 之间通过网络 3 建立 TCP 连接。该终端 1 的具体实例包括但并不限于台式计算机、便携式计算机、智能手机、平板电脑或者其他类似的运算装置。网络 3 可为任意的网络连接方式,例如互联网(Internet)、移动互联网(如电信运营商提供的 2G、3G 网络)、局域网(有线或者无线)等。

[0027] 参阅图 3 所示,本发明第一实施例提供一种 TCP 连接的建立方法,该 TCP 连接的建立方法包括以下步骤:

[0028] 步骤 S1,截获该客户端 10 发出的用于请求建立与服务器 2 之间的传输控制协议 TCP 连接的同步 SYN 包;

[0029] 步骤 S2,根据该 SYN 包,模拟服务器 2 的代理机制构造一同步确认 SYNACK 包,将该 SYNACK 包返回至该客户端 10,使客户端 10 根据该 SYNACK 包发送一确认 ACK 包至该服务器 2,从而与该服务器 2 建立 TCP 连接。

[0030] 按照上述的 TCP 连接的建立方法,通过截获客户端 10 发送的 SYN 包,并模拟服务器 2 向客户端 10 返回 SYNACK 包,从而建立客户端 10 与服务器 2 之间的 TCP 连接,可以减少 TCP 连接建立过程中客户端 10 与服务器 2 之间握手交互的次数,提高建立 TCP 连接的速度,并降低服务器的内存消耗。

[0031] 在一些实例中,上述方法的各步骤的实现细节如下:

[0032] 步骤 S1 所述的 SYN 包中 TCP 首部的 SYN 标志位被置 1,表示请求与服务器 2 建立 TCP 连接。此外,该 SYN 包还携带有一个序列号(sequence number) x。该 SYN 包中的序列号 x 可以由客户端 10 按照一定的规律计算得到,或者随机产生,例如该 x=1234567。

[0033] 步骤 S2 所述的服务器 2 的代理机制例如为 Linux 系统支持的 syn-cookies 机制。若服务器 2 不通过该代理机制而直接接收到该 SYN 包,将检查该 SYN 包中 TCP 首部的 SYN 标志位是否为 1,若是,则确认客户端 10 要求与服务器 2 建立 TCP 连接,并向客户端 10 返回一个 SYNACK 包。该 SYNACK 包中 TCP 首部的 SYN 标志位和 ACK 标志位都被置 1,且该 SYNACK 包包括一个确认号(acknowledge number)y 和序列号 z。该确认号 y 是在该 SYN 包中获取的序列号 x 的基础上加 1 得到的,因此, y=x+1=1234568。该序列号 z 通常由服务器 2 按照一定的规律计算得到或随机产生。

[0034] 但是,若服务器 2 启用该代理机制,例如所述 syn-cookies 机制,则该代理机制将代理服务器 2 接收该 SYN 包。若该代理机制确认该 SYN 包中 TCP 首部的 SYN 标志为 1,则也会构造一个包括一个确认号 y 和一个序列号 a 的 SYNACK 包,并将所构造的 SYNACK 包返回至客户端 10。该 SYNACK 包中 TCP 首部的 SYN 标志位和 ACK 标志位都被置 1。该确认号 y 同样由所述序列号 x 加 1 得到,而该序列号 a 可以通过使用一预定算法,例如通过对该 SYN 包的源地址(客户端 10 的 IP 地址)、目标地址(服务器 2 的 IP 地址)、源端口、目标端口等数据进行加密运算,例如哈希(Hash)加密运算得到,该序列号 a 例如为 7654321。

[0035] 该代理机制一般被嵌入在服务器 2 的 Linux 系统的内核中实现,在本实施例中,可以通过修改内核源代码的形式将服务器 2 的该代理机制暴露出来,以模拟服务器 2 的该代理机制。例如,可以将服务器 2 的该代理机制同步到终端 1 或另一服务器中。将该代理机制同步到终端 1 或所述另一服务器中的方法可以是将服务器 2 的该代理机制的源代码写入终端 1 或所述另一服务器中以执行。因此,当步骤 S1 截获客户端 10 发出的该 SYN 包后,步骤 S2 可以模拟服务器 2 的该代理机制构造该 SYNACK 包,即构造该 SYNACK 包的方法与上一

段中所述的服务器 2 的代理机制构造该 SYNACK 包的方法相同, 该 SYNACK 包中 TCP 首部的 SYN 标志位和 ACK 标志位也都被置 1, 且该 SYNACK 包中也包括所述确认号 y 和所述序列号 a。然后, 步骤 S2 将所构造的该 SYNACK 包返回至客户端 10。

[0036] 客户端 10 接收到该 SYNACK 包后, 若检查到该 SYNACK 包中 TCP 首部的 SYN 标志位和 ACK 标志位都被置 1, 则验证该 SYNACK 包是否合法。具体而言, 客户端 10 将检查该 SYNACK 包中的确认号 y 是否等客户端 10 发出的 SYN 包中序列号 x 加 1, 若是, 则客户端 10 验证该 SYNACK 包合法, 然后向服务器 2 发送一个 ACK 包, 该 ACK 包中 TCP 首部的 ACK 标志位被置 1, 且该 ACK 包携带有确认号 q。该确认号 q 是在该 SYNACK 包中序列号 a 的基础上加 1 得到的, 即 $q=a+1=7654322$ 。此外, 客户端 10 还可将待发送数据, 例如基于 HTTP (Hypertext transfer protocol, 超文本传输协议) 的请求(Request) 数据与该 ACK 包共同发送至服务器 2。

[0037] 服务器 2 的代理机制代理服务器 2 接收到该 ACK 包后, 若检查到该 ACK 包中 TCP 首部的 ACK 标志位被置 1, 则验证该 ACK 包是否合法。具体而言, 服务器 2 的该代理机制将按照上述的步骤 S2 模拟服务器 2 的代理机制构造 SYNACK 包时生成序列号 a 的方法, 即使用所述的预定算法, 例如对该 ACK 包的源地址(客户端 10 的 IP 地址)、目标地址(服务器 2 的 IP 地址)、源端口、目标端口等数据进行加密运算, 得到一个数值, 称之为 cookie。然后, 服务器 2 的该代理机制判断该 ACK 包中的确认号 q 减 1 后, 是否与该 cookie 相等。由于服务器 2 的代理机制生成该 cookie 的方法和终端 1 模拟服务器 2 的代理机制生成序列号 a 的方法相同, 所以该 cookie 与序列号 a 也应当相等。而确认号 $q = a+1$, 因此, 该确认号 q 减 1 后, 与该 cookie 也应当相等。若服务器 2 的该代理机制判断该确认号 q 减 1 后与该 cookie 相等, 则验证该 ACK 包合法, 服务器 2 与客户端 10 成功建立 TCP 连接。

[0038] 综上所述, 由于该 SYN 包被截获而未发送至服务器 2, 通过模拟服务器 2 向客户端返回 SYNACK 包, 使三次握手过程的前两次都未经过客户端 10 与服务器 2 的交互过程, 从而减少 TCP 连接建立过程中客户端 10 与服务器 2 之间握手交互的次数, 可以提高建立 TCP 连接的速度。此外, 由于服务器 2 未接收到该 SYN 包, 因此无需分配内存空间专门存储接收到该 SYN 包后产生的半连接状态的条目, 从而可以降低服务器 2 的内存消耗。

[0039] 参阅图 4 所示, 本发明第二实施例提供一种 TCP 连接的建立装置 100, 其包括截获模块 101 和构造模块 102。可以理解, 上述的各模块是指计算机程序或者程序段, 用于执行某一项或多项特定的功能。此外, 上述各模块的区分并不代表实际的程序代码也必须是分开的。

[0040] 截获模块 101, 用于截获该客户端 10 发出的用于请求建立与服务器 2 之间的传输控制协议 TCP 连接的同步 SYN 包。该 SYN 包携带有一序列号 x。

[0041] 构造模块 102, 用于根据该 SYN 包, 模拟服务器 2 的代理机制构造一同步确认 SYNACK 包, 将该 SYNACK 包返回至该客户端 10, 使客户端 10 根据该 SYNACK 包发送一确认 ACK 包至该服务器 2, 从而与该服务器 2 建立 TCP 连接。该服务器 2 的代理机制例如为 Linux 系统支持的 syn-cookies 机制。具体而言, 构造模块 102 通过对该序列号 x 加 1 生成一确认号 y, 并使用一预定算法生成一序列号 a, 然后生成携带该确认号 y 和序列号 a 的该 SYNACK 包, 并将该 SYNACK 包返回至该客户端 10。参阅图 5 所示, 为客户端 10 与服务器 2 通过该 TCP 连接的建立装置 100 建立 TCP 连接的交互过程示意图。

[0042] 客户端 10 接收到该 SYNACK 包后, 将先根据该确认号 y 验证该 SYNACK 包是否合法。若该确认号 y 等于该序列号 x 加 1, 则验证该 SYNACK 包合法, 然后向服务器 2 发送该 ACK 包。该 ACK 包携带有一确认号 q。该确认号 q 应当等于序列号 a 加 1。此外, 客户端 10 还可将待发送数据与该 ACK 包共同发送至服务器 2。

[0043] 服务器 2 接收到该 ACK 包后, 将根据该确认号 q 通过所述代理机制验证该 ACK 包是否合法。具体而言, 服务器 2 的该代理机制使用所述预定算法生成一个数值, 然后判断该确认号 q 减 1 后与该数值是否相等, 若是, 则验证该 ACK 包合法, 服务器 2 与客户端 10 成功建立 TCP 连接。

[0044] 对于以上各模块的具体工作过程, 可进一步参考本发明第一实施例提供的 TCP 连接的建立方法, 在此不再重复。

[0045] 综上所述, 本实施例的 TCP 连接的建立装置 100, 通过截获客户端 10 发送的 SYN 包, 并模拟服务器 2 向客户端 10 返回 SYNACK 包, 从而建立客户端 10 与服务器 2 之间的 TCP 连接, 可以减少 TCP 连接建立过程中客户端 10 与服务器 2 之间握手交互的次数, 提高建立 TCP 连接的速度, 并降低服务器的内存消耗。

[0046] 参阅图 6 所示, 本发明第三实施例提供所述的终端 1, 该终端 1 包括本发明第二实施例提供的 TCP 连接的建立装置 100。通过终端 1 中的该 TCP 连接的建立装置 100, 可以在服务器 2 与终端 1 中的客户端 10 之间通过网络 3 建立 TCP 连接。

[0047] 参阅图 7 所示, 本发明第四实施例提供一种系统 200, 该系统 200 包括至少一个所述服务器 2 和至少一个本发明第三实施例提供的所述终端 1。通过该系统 200 可以在服务器 2 与终端 1 中的客户端 10 之间通过网络 3 建立 TCP 连接。

[0048] 参阅图 8 所示, 本发明第五实施例提供一种服务器 30, 该服务器 30 包括本发明第二实施例提供的 TCP 连接的建立装置 100。所述服务器 30 通过网络 3 分别与第一实施例或第二实施例中所述的终端 1 和服务器 2 连接。通过服务器 30 可以在服务器 2 与第一实施例或第二实施例中所述的终端 1 中的客户端 10 之间通过网络 3 建立 TCP 连接。

[0049] 参阅图 9 所示, 本发明第六实施例提供一种系统 300, 该系统 300 包括至少一个第一实施例或第二实施例中所述的终端 1, 至少一个所述服务器 2, 所述至少一个终端 1 和至少一个服务器 2 通过至少一个本发明第五实施例提供的服务器 30 连接。通过该系统 300 可以在服务器 2 与第一实施例或第二实施例中所述的终端 1 中的客户端 10 之间通过网络 3 建立 TCP 连接。

[0050] 此外, 本发明实施例还提供一种计算机可读存储介质, 其内存储有计算机可执行指令, 上述的计算机可读存储介质例如为非易失性存储器例如光盘、硬盘、或者闪存。上述的计算机可执行指令用于让计算机或者类似的运算装置完成上述的 TCP 连接的建立方法中的各种操作。

[0051] 以上所述, 仅是本发明的较佳实施例而已, 并非对本发明作任何形式上的限制, 虽然本发明已以较佳实施例揭示如上, 然而并非用以限定本发明, 任何本领域技术人员, 在不脱离本发明技术方案范围内, 当可利用上述揭示的技术内容做出些许更动或修饰为等同变化的等效实施例, 但凡是未脱离本发明技术方案内容, 依据本发明的技术实质对以上实施例所作的任何简介修改、等同变化与修饰, 均仍属于本发明技术方案的范围内。

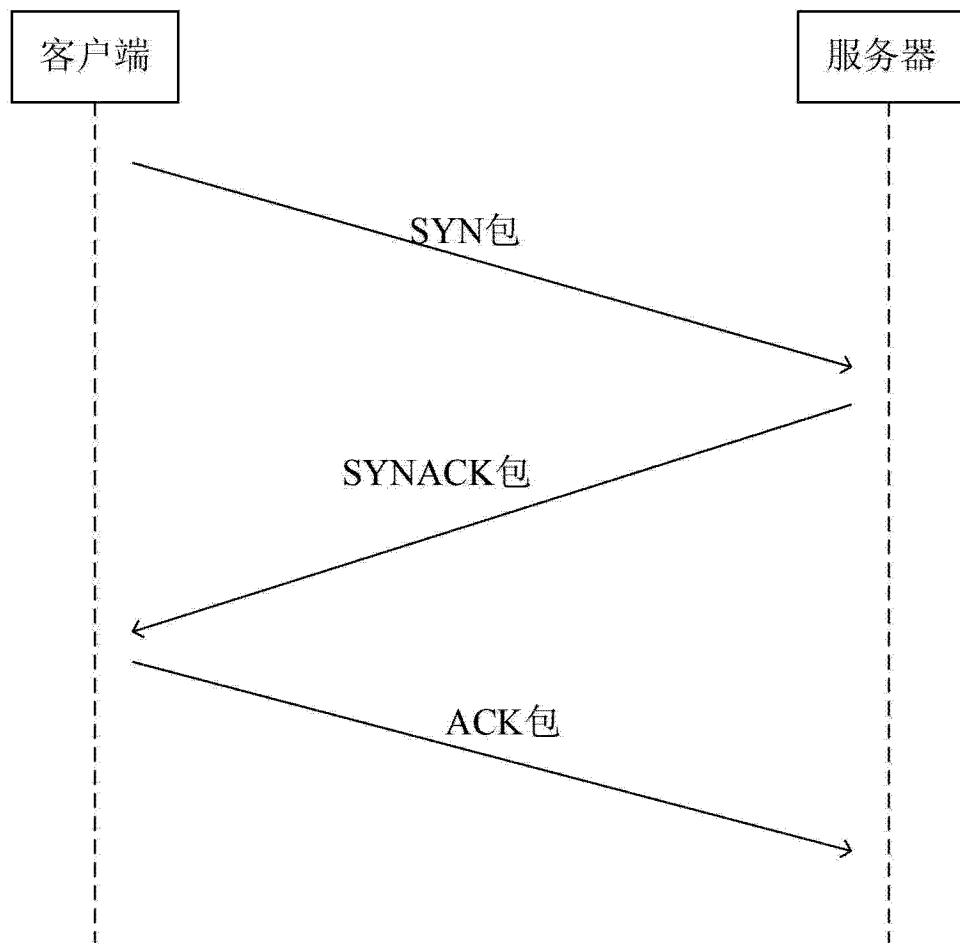


图 1

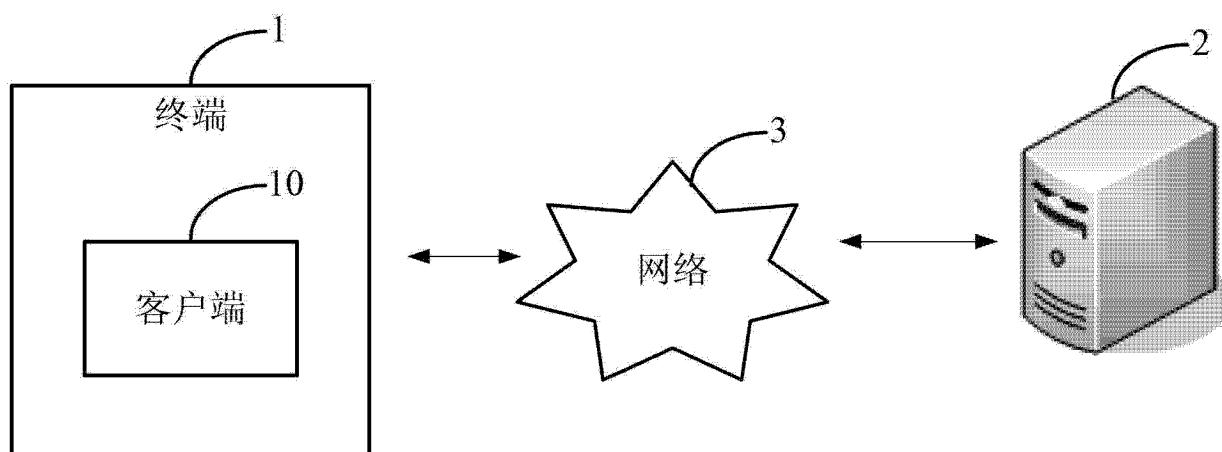


图 2

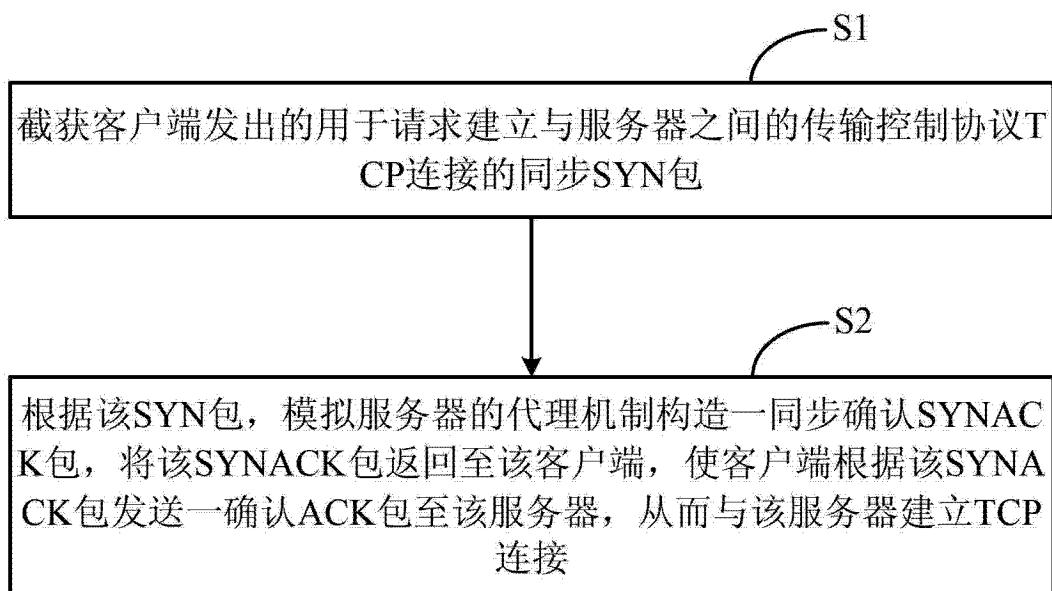


图 3

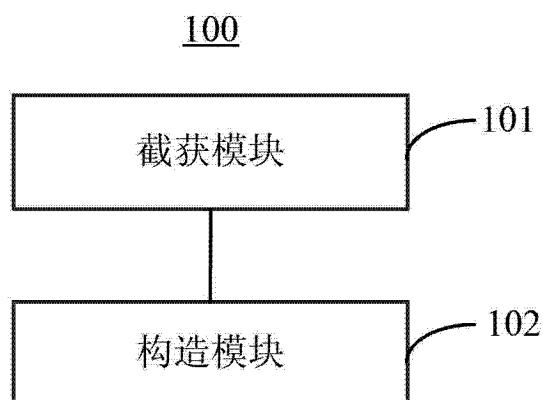


图 4

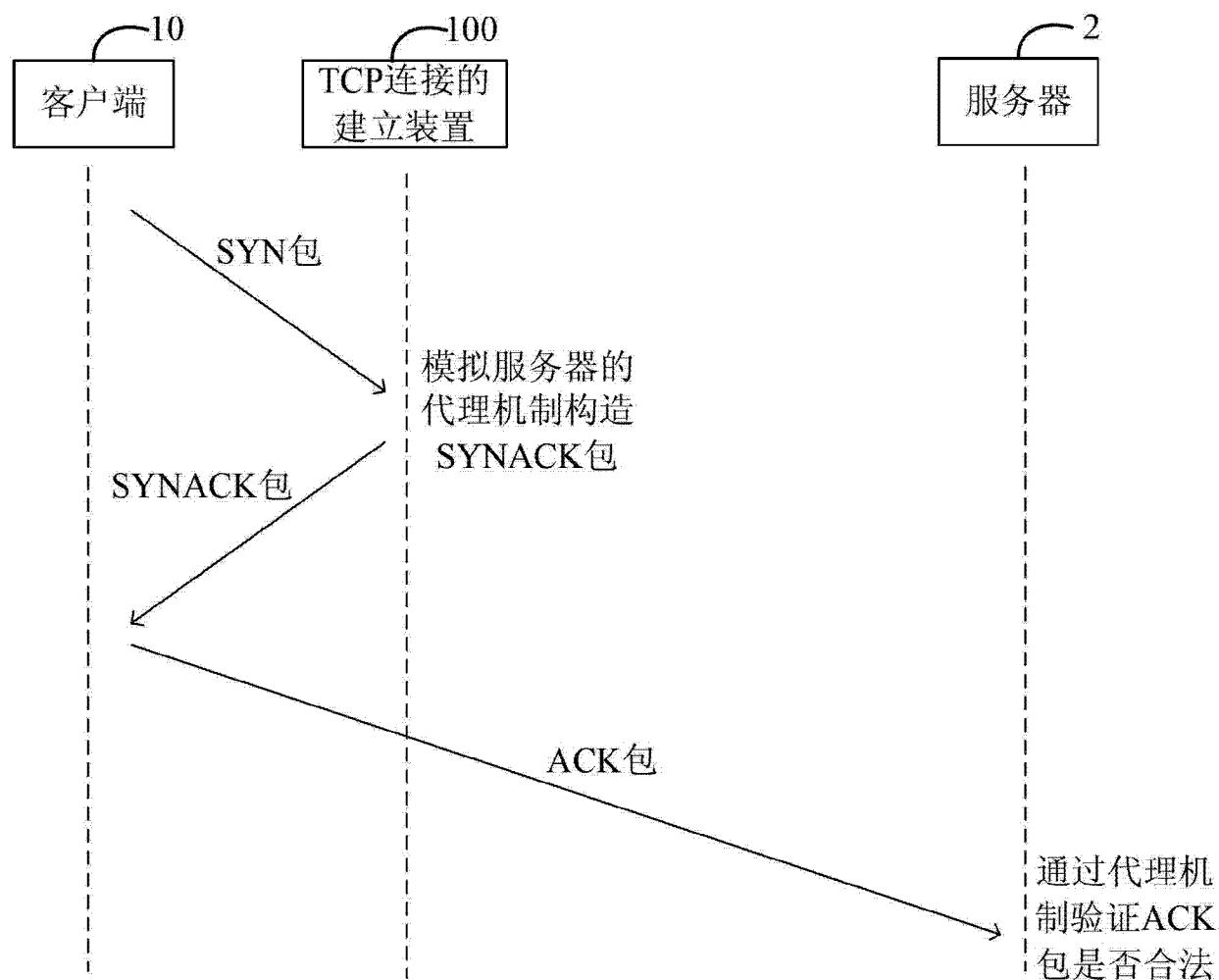


图 5

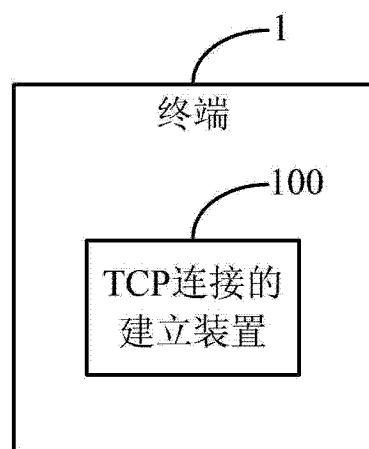


图 6

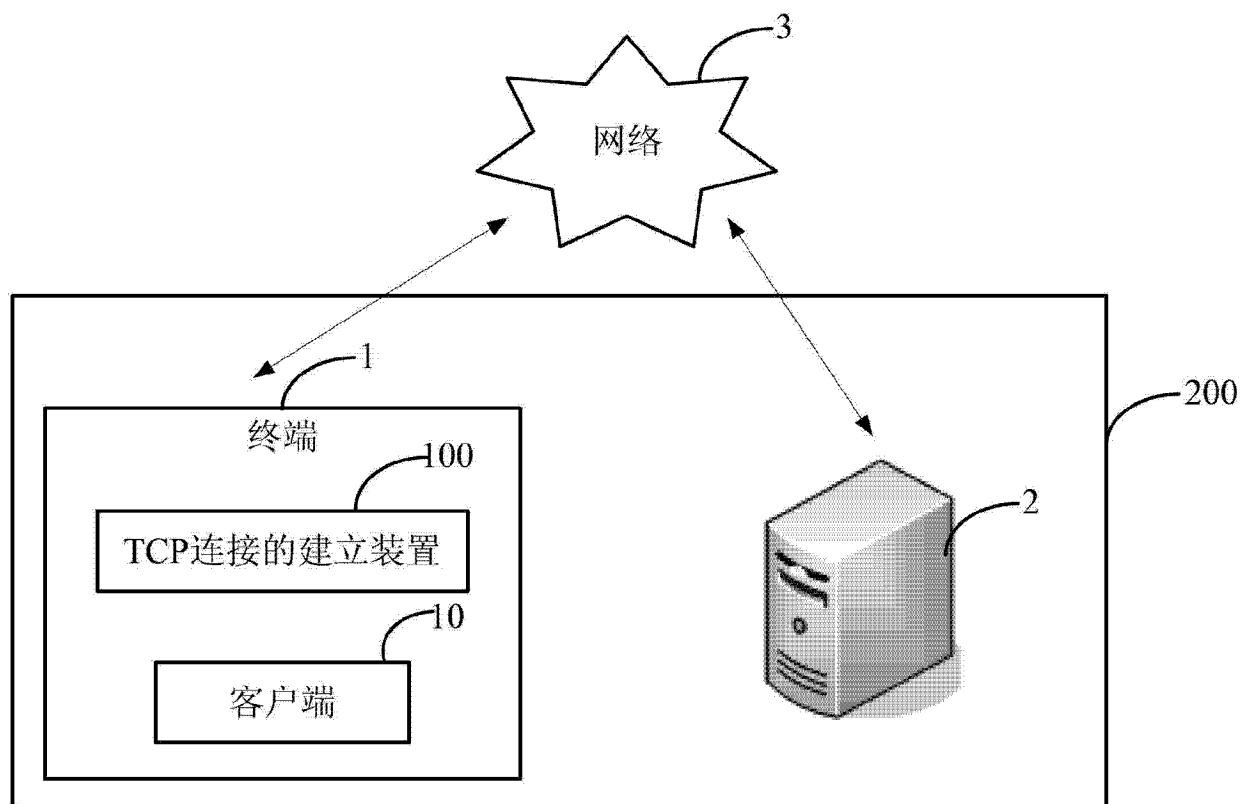


图 7

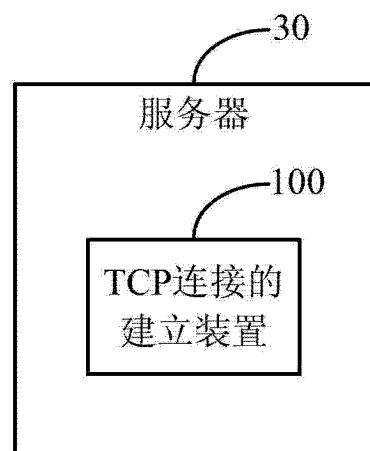


图 8

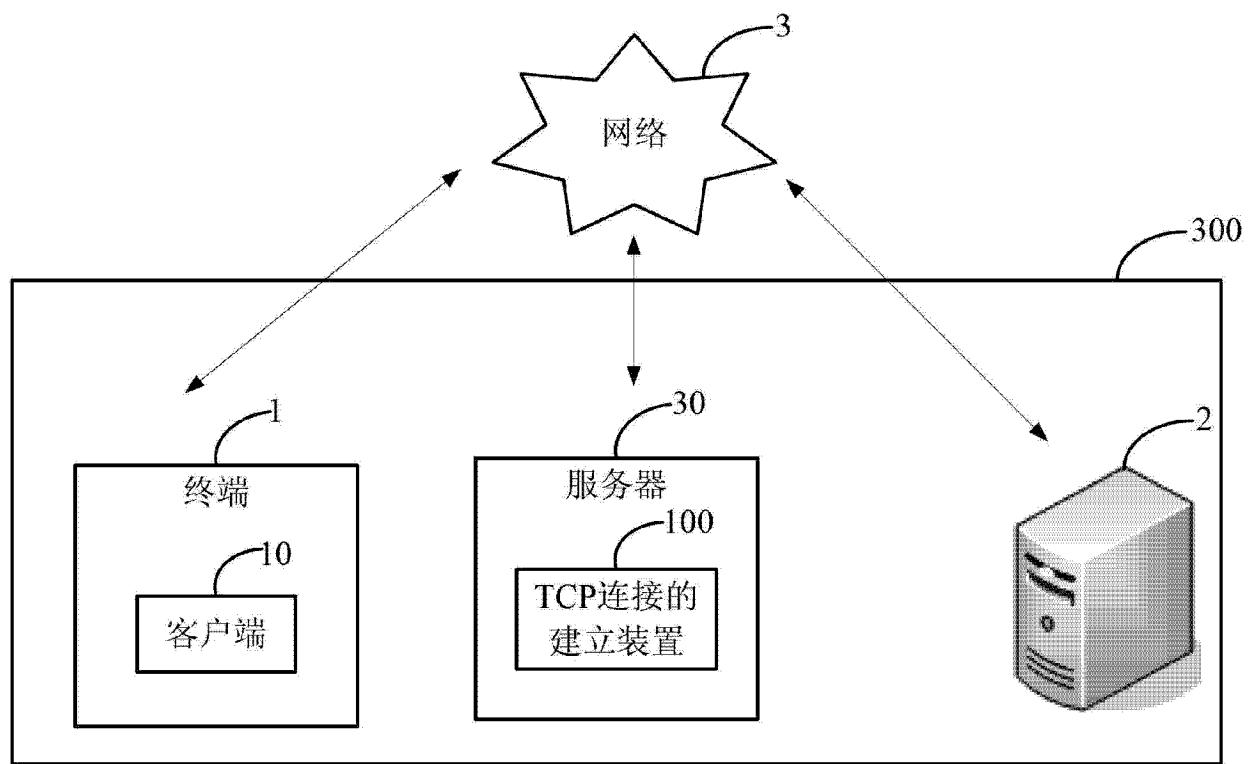


图 9