doi:10.3772/j.issn.1002-0470.2017.02.002

一种适于高可靠环境的可信网络构建方法①

谭 励②* 张亚明* 杨明华** 胡计鹏*

(*北京工商大学计算机与信息工程学院 北京 100048)

(**火箭军装备研究院第四研究所 北京 100094)

摘 要 为了保证网络接入终端的可信性和网络安全,结合可信网络连接技术以及高可用(HA)集群思想,提出了一种适于高可靠环境的可信网络构建方法。该方法在终端接入网络时,会根据平台身份是否合法,软件版本是否符合标准,是否接入了非法外设,以及网络端口的状态等安全属性进行可信准入判定,同时服务器端和客户端采用双冗余热备方式,提高系统的可靠性。实验测试表明,该方法能够成功实现服务的可信切换和终端的可信接入控制。

关键词 可信网络连接, 高可用(HA), 集群, 双机热备, 可信计算

0 引 言

随着计算机网络的日益发展,人类与信息化网络的结合越来越密切,但是也存在由于网络的不可靠性给人们带来的巨大困扰。2009 年 5 月,美国总统奥巴马公布了安全部门给出的网络空间安全威胁评估报告[1],报告中指出网络安全威胁是美国面临的严重国家安全挑战,在中国,截至 2015 年底,共处置 81256 起网络安全事件,比 2014 年增长 124%。在 2016 年 9 月份召开的抗恶劣会议中蒋晓原研究员提出未来计算机必须自主可控,要求未来国产计算机提供统一可信计算服务支撑能力。文献[2]提及的可信网络,正是针对这种情况而提出的一种解决方案,它是从体系结构和基础协议入手进行彻底创新,旨在通过提供一致的安全服务体系结构来为网络提供安全性保障。

2003 年可信计算组织(trusted computing group, TCG)成立,可信计算技术得到了迅速发展。人们已经意识到,在面对现有各种安全风险与威胁时,不仅需要自顶向下的安全体系设计,还需要从终端开始自底向上地保证计算系统的可信,不仅要保证终端计算环境的可信,还要把终端计算环境的可信扩展到网络,使得网络成为一个可信的计算环境[3]。目前国际上许多国家和研究机构都对网络安全接入技术做了大量研究,其中最有影响力的架构有以下三种:可信计算组织(trusted computing group, TCG)的可信网络连接(trusted network connect, TNC[4])架构,思科的网络接入控制(Cisco network access control, NAC)架构,微软的网络访问保护(network access protection, NAP)架构。NAC 与 NAP 的标准是不开放的,对于推广这样的标准存在很大的问题。TCG 工作组制定的 TNC 标准,是一套开放的标准网络接入控制架构,旨在跨越多个平台、外围和产品,为支持硬件的可信计算和安全技术(包括硬件构件和软件界面等)提供开放的标准。

可信网络连接(TNC)是对可信平台应用的扩展,也是可信计算机制与网络接入控制机制的结合。其核心思想是在终端接入网络之前,首先对用户的身份进行认证;如果用户身份认证通过,继续对终端

① 国家核高基重大专项(2014ZX01040501-002),北京市自然科学基金(4172013),国家自然科学基金(61402022)和北京市哲学社科规划项目(14JGB033)资助。

② 女,1980 年生,教授,博士;研究方向:无线传感器网络,智能信息处理;联系人,E-mail: tanli@th.btbu.edu.cn

(收稿日期:2016-07-12)

平台的身份进行认证;如果平台身份认证通过,继续对终端平台的可信状态进行度量,如果度量结果满足网络接入的安全策略,则允许终端接入网络,否则将终端连接到指定的隔离修复区域,对其进行安全性修补和升级[5]。TNC 旨在将终端的可信状态延续到网络中,使信任链从终端扩展到网络,它是网络接入控制的一种实现方式,是一种主动性的防御方法,能够将大部分的潜在攻击在发生之前进行抑制。

目前在电力、高铁、石油等领域,对服务的安全性、可用性有相当高的要求,一方面需要保证服务的全天候不间断,另一方面需要保证网络的绝对安全。本文正是在这种环境下,结合 TNC 技术和高可用(high availability,HA)集群中的双机热备(hot-stand by)思想[6],将可信网络技术扩展到这种高安全、高可靠环境中,通过服务器端、客户端冗余设计和心跳检测设计,实时检验主备双方的活动状态和可信状态,当其中一方不符合安全策略要求或出现软硬件故障时,则由另一方立即接替其执行相关任务。

1　相关研究

虽然可信网络连接(TNC)只是一种技术架构,但是它开创性地提出了将可信计算机制引入网络,引起了国内外研究者对此更加深入和广泛的研究,其中可信网络连接与可信网络体系结构的研究是其中较为重要的一方面。2007 年 2 月,北京工业大学牵头组织可信计算关键标准的研究,提出了三元、三层、对等、集中管理的可信网络连接架构(TCA),通过引入一个策略管理器作为第三方,对访问请求者和访问控制器进行集中管理。网络访问控制层和可信平台评估层执行基于策略管理器为可信第三方的三元对等鉴别协议,实现访问请求者和访问控制器之间的双向用户认证和双向平台可信性评估。该架构采用国产自主知识产权的鉴别协议,将访问请求者和访问控制器作为对等实体,以策略管理器为可信第三方,既简化了身份管理、策略管理和证书管理机制,又保证了终端与网络的双向认证,具有很大的创新性[5]。2007 年 6 月,颜菲等人针对网络认证终端缺乏安全保护等问题提出了一种基于 TNC 的安全认证协议,该协议在可信环境下把终端完整性度量和 PKI 技术相结合,确保了通信双方的平台完整性和终端、网络两者之间的认证安全[7]。2009 年 1 月,Rehbock 等人提出一种在开放网络环境下的安全技术来保证数据的可靠传输,该技术主要采用安全声明标记语言(SAML),使用安全卡来封装完整性信息数据,保证数据的安全传输[8]。2010 年方娟等人提出了一种基于可信度的层次化评估模型,将可信度按照不同的影响因素进行分层,使用层次分析法对可信度进行度量分析[9]。2013 年肖悦雷等人提出了一种基于三元对等鉴别的可信网络体系结构,详细定义了 TCA 的接口协议,通过对 TCA 及其接口协议的分析,证明了 TCA 不但可以防止不符合安全策略的终端接入网络,而且还可以保障可信网络连接自身的安全性,具有良好的应用扩展性[10]。沈昌祥院士在密码学报中提出一种基于国产密码算法的多层可信体系,其体系结构以国产密码体系为基础,以可信平台控制模块为信任根,以可信主板为平台,以软件为核心,通过可信网络作为纽带整合各种应用[11]。2015 年 12 月金雷等人提出了一个基于可信网络连接框架的网络终端认证模型 TNTAM,该模型通过加入身份认证系统模块、信息访问鉴别模块(information access authentication module, IADM)和引入策略管理器来加强终端认证的安全性,实现了用户、改进的认证智能卡以及可信终端三者间的相互认证,并确保了可信终端请求网络服务的可信性和通信安全[12]。

综上所述,目前已有的研究成果大多专注于可信网络的二元、三元对等模型及接口协议研究,针对应用领域,尤其是工控系统中普遍采用的冗余体系架构,已有的模型、方法和协议缺少相应的支撑,还需要进行深入的研究。

2　高可靠环境下的可信网络构建方法

2.1　总体设计

高可靠环境下的可信网络结合了可信网络连接(TNC)和双机热备(hot-stand by)技术,是一种主动性的防御方法,能有效防御潜在的攻击,提高整个网

络的安全性、可靠性。其核心思想是:利用可信密码模块(trusted cryptography module, TCM)芯片的安全保护特性,在终端设备可信启动的基础上,完成终端和可信网络接入点之间的身份和可信性验证,依据验证结果,可信网络接入点做出相应判断,将终端排除在可信网络之外,或是根据其可信的不同程度,将终端归入网络的不同信任域中。其中服务端、客户端采用双冗余设计,通过可信、冗余设计提高应用系统的安全性、可靠性。高可靠环境下的可信网络连接架构如图 1 所示,它由 5 元、3 层组成,其中 5 元包括 2 个访问请求者、1 个访问控制器和 2 个策略管理器,3 层分别是网络访问层、完整性评估层以及完整性度量层。

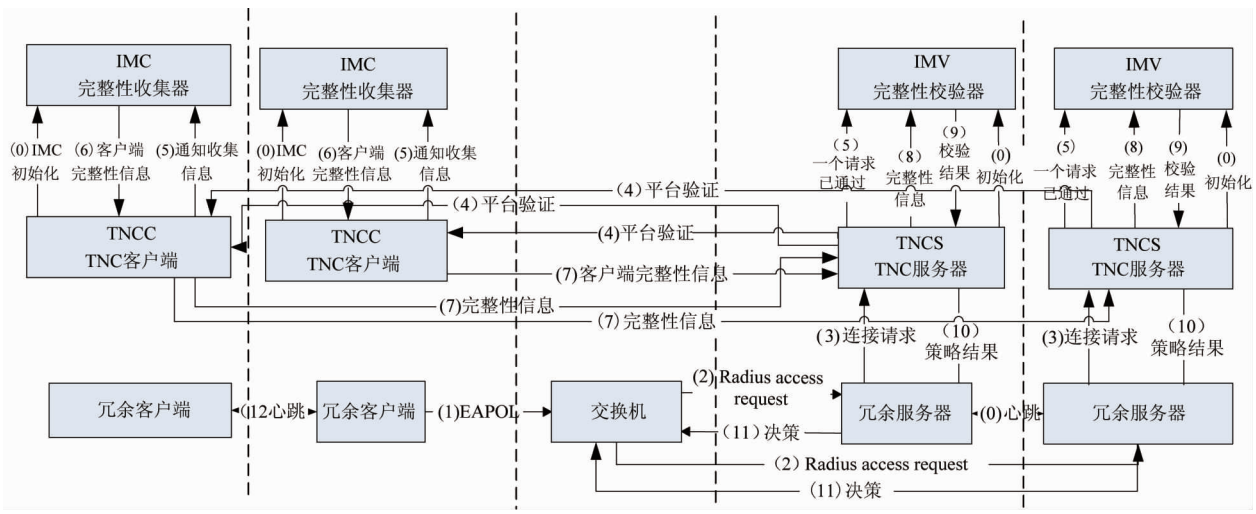

图 1　高可靠环境下的可信网络连接架构

图 1 中,TNCC 是可信网络连接客户端,TNCS 是可信网络连接服务器,IMC 是完整性测量收集者,IMV 是完整性测量校验者。具体实现流程描述如下:

(0) 在进行网络连接和平台完整性验证之前,TNCC 需要对每一个 IMC 进行初始化。同样,TNCS 也要对 IMV 进行初始化。服务器端的两台冗余机启动心跳检测,如果主用服务器一旦出现故障或不被信任则执行主备切换,保证服务持续不间断。

(1) 冗余客户端发起认证,发送基于局域网的扩展认证协议(extensible authentication protocol over LAN, EAPOL)包给交换机,请求接入。

(2) 交换机会回应一个 Challenge 包,要求客户端提供用户名,服务器根据收到的用户名与本地数据库进行对比,找到对应的密码,用随机生成的密码字对该密码进行加密码,并且把该密码字发送到客户端,让客户端对其密码进行加密,然后经过交换机传送到服务器端,与本地已经加密过的密码进行匹配,如果比对成功,则认为用户名和密码符合要求,然后继续后续的完整性认证。客户端发送 Radius-Access-Request 请求报文,交换机对客户端发来的数据包进行封包,向当前可用的认证服务器发送请求接入包。

(3) 当前认证服务器通知相应的 TNCS 有一个请求。

(4) TNCS 与 TNCC 交互,进行平台身份验证,如果验证通过,继续下面的验证。

(5) 在冗余终端方,TNCC 通知 IMC 收集客户端的完整性信息,在服务器端,当前 TNCS 通知自己的 IMV 校验器有请求已通过用户名和密码认证,让其准备接受终端发来的数据对其进行校验。

(6) IMC 收集信息之后发送给 TNCC。

(7) TNCC 把由 IMC 收集到的客户端完整性信息通过接口发送给当前 TNCS。

(8) 当前 TNCS 收到 IMC 传来的完整性信息后交给自己的 IMV 校验器进行校验。

(9) IMV 对完整性信息校验后,把校验结果回送给 TNCS,由 TNCS 进行评估,做出一个推荐操作。

(10) 如果检验通过,会通知到当前冗余服务器进行最终决定。

(11) 当前认证服务器做出决策后通知交换机执行最终决定,根据交换机的设置,如果通过就进入到可信区。如果不通过就进入隔离区进行修复,修复通过之后可重新申请进入可信区。

(12) 认证通过之后,冗余终端之间开始执行心跳检测,如果其中一台终端因为某种原因导致不可信从而被断开网络,则另一台客户端继续为用户提供服务。

(13) 等待一个时间节拍,重新开始认证过程,返回步骤(1)。

其中 IMC-IMV 完整性检测内容包括:(1)端口度量,检查端口状态,保证没有私自打开的后门;(2)外设度量,检查是否接入了 USB 移动存储设备、USB 光驱等;(3)杀毒软件是否安装,版本是否符合要求;(4)TCM(可信密码模块)对平台的可信度量结果。满足安全策略要求的终端可直接访问网络资源;否则,首先对终端进行隔离修复,待满足安全策略要求后才可访问网络资源,对无法修复的终端则拒绝其对网络资源的访问。

2.2　协议设计

客户端认证过程遵循 802.1X 协议扩展的 EAPOL(基于局域网的扩展认证协议),其中终端 1 和终端 2 都需要进行可信认证。终端都认证通过之后二者之间执行心跳检测程序,检测对方状态。服务器端主备之间同样执行心跳程序进行完整性和可信性检测。冗余系统可信网络连接协议认证过程中如图 2 所示。

主要认证流程如下:

(1) 终端 1 发起认证,发送 EAPOL 包给交换机,请求接入。

(2) 终端 2 发起认证,发送 EAPOL 包给交换机,请求接入。

(3) 交换机会回应终端 1 一个扩展认证协议(EAP)-Request/Identity 包,让终端 1 把用户名报上来。

(4) 交换机会回应终端 2 一个 EAP-Request/Identity 包,让终端 2 把用户名报上来。

(5) 终端 1 回应一个 EAP-Response/Identity1 给接入设备的请求,其中包括用户名。

(6) 终端 2 回应一个 EAP-Response/Identity2 给接入设备的请求,其中包括用户名。

(7) 接入设备将 EAP-Response/Identity1 报文封装到 Radius-Access-Request1 报文中,发送给认证服务器,服务器之间执行心跳检测程序,如果一台服务器故障或状态不可信,则另一台服务器继续接替工作。

(8) 接入设备将 EAP-Response/Identity2 报文封装到 Radius- Access-Request2 报文中,发送给认证服务器,服务器之间执行心跳检测程序,如果一台服务器故障或状态不可信,则另一台服务器继续接替工作。

(9) 认证服务器产生一个 Challenge1,通过接入设备将 Radius-Access-Challenge1 报文发送给接入设备,其中包含有 EAP-Request/MD5-Challenge1。

(10) 认证服务器产生一个 Challenge2,通过接入设备将 Radius-Access-Challenge2 报文发送给接入设备,其中包含有 EAP-Request/MD5-Challenge2。

(11) 接入设备通过 EAP-Request/MD5-Challenge1 发送给终端 1,要求客户端进行认证。

(12) 接入设备通过 EAP-Request/MD5-Challenge2 发送给终端 2,要求客户端进行认证。

(13) 终端 1 收到 EAP-Request/MD5-Challenge1 报文后,将密码和 Challenge1 做 MD5 运算后的 Challenged-Password1 通过 EAP-Response/MD5-Challenge1 回应给接入设备。

(14) 终端 2 收到 EAP-Request/MD5-Challenge2 报文后,将密码和 Challenge2 做 MD5 运算后的 Challenged-Password2 通过 EAP-Response/MD5-Challenge2 回应给接入设备。

(15) 接入设备将 Challenge1,Challenged Password1 和用户名封装成 Radius-Access-Request1 包一起送到认证服务器,由认证服务器进行认证。

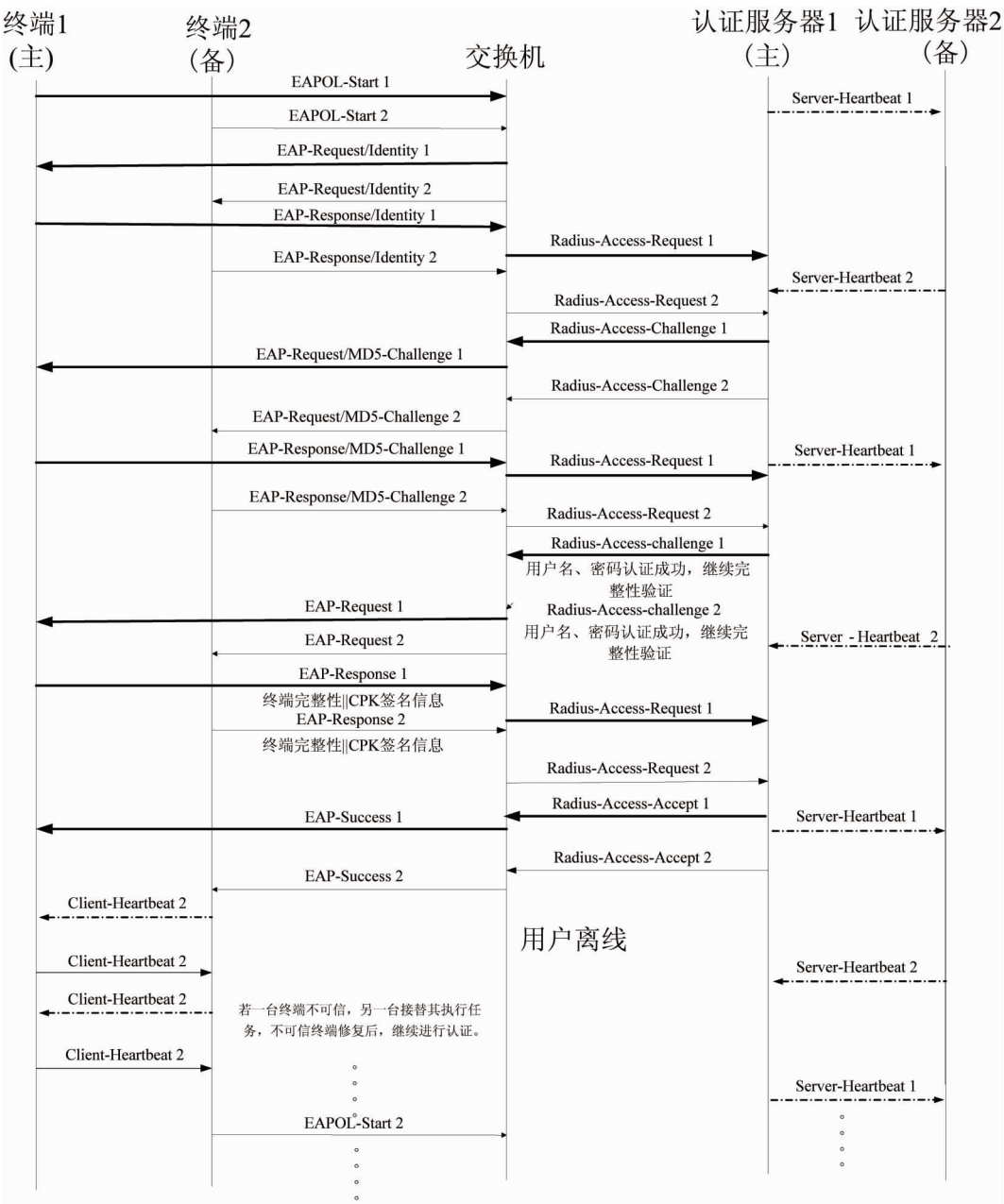

图 2　冗余系统可信网络连接协议

（16）接入设备将 Challenge2，Challenged Password2 和用户名封装成 Radius-Access-Request2 包一起送到认证服务器，由认证服务器进行认证。

（17）服务器收到 Challenge Password1 之后，会和本地经过加密运算后的密码进行对比，如果比对成功，服务器会继续根据服务器端预制策略要求终端 1 继续发送终端完整性信息。

（18）服务器收到 Challenge Password2 之后，会和本地经过加密运算后的密码进行对比，如果比对成功，服务器会继续根据服务器端预制策略要求终端 2 继续发送终端完整性信息。

（19）交换设备收到 Challenge1 之后，要求终端 1 按照服务器策略要求收集完整性信息。

（20）交换设备收到 Challenge2 之后，要求终端 2 按照服务器策略要求收集完整性信息。

（21）终端 1 中的 IMC 收集器收集到信息之后将数据发送到交换机。

（22）终端 2 中的 IMC 收集器收集到信息之后将数据发送到交换机。

（23）交换机收到终端 1 发来的完整性信息之

后,转发给服务器端。

(24) 交换机收到终端 2 发来的完整性信息之后,转发给服务器端。

(25) 根据服务器预定策略,检测终端 1 信息是否符合,决定是否允许终端接入网络。

(26) 根据服务器预定策略,检测终端 2 信息是否符合,决定是否允许终端接入网络。

(27) 如果服务器端策略匹配成功,允许终端 1 接入,发送给客户端 EAP-Success1 消息。

(28) 如果服务器端策略匹配成功,允许终端 2 接入,发送给客户端 EAP-Success2 消息。

(29) 自此终端之间开始执行心跳检测程序,时刻监视对方状态。

(30) 如果客户端因为某些原因导致不可信,不能访问网络资源,则由另一台继续执行相关应用程序为用户提供服务。

2.3　冗余设计

通过网络寻求服务已成为信息服务的主要方式,由于任何一个服务停顿都可能造成不可估量的损失和影响,因此许多行业都要求能够提供不间断的服务。对于可信网络连接技术,则必须保证服务器端认证服务的持续运行,不会因为服务器端在某一时刻无法工作而无法完成对客户端的认证;而客户端同样需要具有持久访问网络的能力,不应因平台故障或状态不可信而无法访问网络资源。

本文基于高可用(HA)集群中的双机热备思想,在服务器端和客户端分别采用冗余设计,如图 3 所示。上层是两台客户机,采用心跳检测机制进行

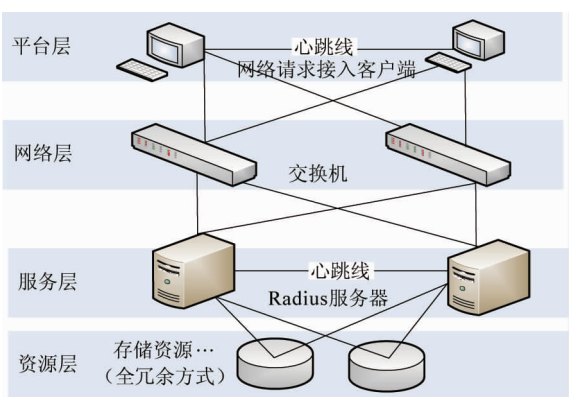

图 3　冗余设计

可用性检测,中间层是支持 802.1X 协议的交换机,下层是两台认证服务器,同样采用心跳进行检测,如果其中一台认证服务器故障或平台不可信,则会立即启动另一台认证服务器继续服务。

在服务器端,每隔一个时间节拍会执行一次心跳检测,其中检测数据包包括平台完整性状态和认证服务程序是否处于运行状态。如果心跳检测失效,那么执行主备切换,由备用认证服务器接替原来的主用服务器继续为网络访问请求接入客户端提供认证服务。

在客户端,两台客户端都认证通过之后,应用业务会在其中一台终端上运行,另一台终端作为备用终端。如果在下一周期接入认证过程中,因为某些原因导致主用终端不再符合服务器端制定的接入策略,将不允许再访问网络,备用终端通过心跳检测机制会探知到主用终端的故障,自己会接替其继续运行应用业务。

服务器端和客户端皆采用全冗余方式,此种冗余方式较之传统主备方式容灾能力更强,能够避免由于存储资源损坏而导致双机热备系统全面崩溃的问题。其中详细的设计流程如图 4 所示。

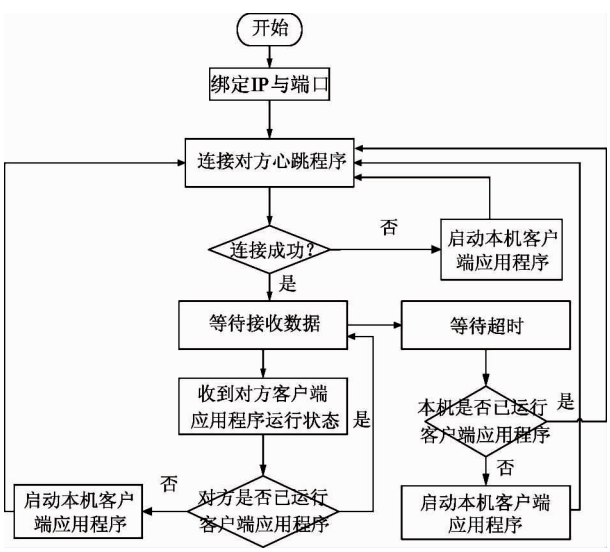

图 4　冗余设计流程图

2.4　EAPOL 包设计

EAPOL 是基于局域网的可扩展认证协议,是基于 802.1X 网络访问认证技术发展而来的[13]。本设计在传统 EAP 包的基础上加上了数据完整性相关

内容,在执行网络接入认证的时候,可以根据服务器端的既定策略进行认证评估。扩展后的 EAP 包如图 5 所示。

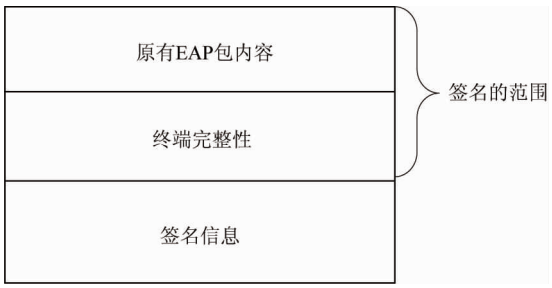

图 5　扩展后的 EAP 包内容

其中数据包格式设计如表 1 所示。

2.5　服务器端可信配置

服务器端安装 Freeradius 之后,需要对其进行相关配置,使其支持 EAP-TNC,然后需要编写配置文件 eap _ config,指定客户端要收集的信息。根据客户端收集到的信息,使用策略(clamavimv. plocy、hostscanner. plocy 等)来进行匹配,如果符合策略就进入 vlan2(可信区),否则进入 vlan3(隔离区)。

表 1　EAP 数据包格式

第 7、8 位	第 6 位	第 5 位	第 4 位	第 3 位	第 2 位	第 1 位
保留	检查当前平台中的核心配置文件是否被修改	当前平台是否接入了非法外设。1 表示接入,0 表示未接入	当前平台的网络端口状态,1 表示打开,0 表示关闭。		XXX 软件版本是否是最近的,0 表示过期,1 表示最新	XXX 软件是否开启,1 表示开启,0 表示没有开启

1. 修改文件/usr/local/etc/raddb/modules/eap,使其支持 EAP-TNC,如下所示:

```
eap{
        #…
        default _ eap _ type = ttls
        #
        tnc{
        }
                #…
                ttls{
                        default _ eap _ type = tnc
                        #
                        use _ tunneled _ reply = yes
```

2. 对于不同访问权限的用户对其进行 vlan 划分,实现资源控制,修改内容如下所示:

```
post-auth {
if (control:TNC-Status  = = "Access") {
        update reply {
            Tunnel-Type : = VLAN
            Tunnel-Medium-Type : = IEEE-802
            Tunnel-Private-Group-ID : = "vlan 0002"
        }
}
elsif (control:TNC-Status  = = "Isolate") {
        update reply {
            Tunnel-Type : = VLAN
            Tunnel-Medium-Type : = IEEE-802
            Tunnel-Private-Group-ID : = "vlan 0003"
        }
}
Post-Auth-Type REJECT {
        # log failed authentications in SQL, too.
        #sql
        attr _ filter. access _ reject
```

3. 修改/etc/tnc _ config 文件,添加 IMV 名称和 IMV 库的路径,库与客户端相对应,要求客户端按照要求收集本机信息到服务器端进行校验,包括杀毒软件,文件完整性,外设等,如下所示:

```
#IMV-Configuration file of TNC@ FHH-TNC-Server
#Example:
IMV "ClamAV" /usr/local/lib/libclamavimv. so. 0. 8. 4
IMV "HostScanner" /usr/local/lib/libhostscannerimv. so. 0. 8. 4
IMV "UsbScanner" /usr/local/lib/libusbscannerimv. so. 0. 8. 4
IMV "File" /usr/local/lib/libclientcheckerimv. so. 0. 8. 4
#IMV "USB" /usr/local/lib/libusbscannerimv. so. 0. 8. 4
```

2.6　客户端可信配置

客户端同样需要进行相关配置,首先修改 wpa _ supplicant 中. config 文件,使其支持 EAP _ TNC,然后修改 tnc _ config 文件,设置 IMC 的库文件路径,与服务器中 IMV 设置相对应。

1. 修改. config 文件:

```
# EAP-TNC and related Trusted Network Connect support (experimental)
CONFIG _ EAP _ TNC = y
#CONFIG _ DRIVER _ NL80211 = y
```

2. 修改 tnc _ config 文件,添加 IMC 名称以及库文件路径,如下所示;

```
#Example:
#IMC "Dummy" /path/to/libDummyIMV. so
#IMC "Dummy" /usr/local/lib/libdummyimv. so. 0. 8. 4
IMC "ClamAV" /usr/local/lib/libclamavimv. so. 0. 8. 4
IMC "HostScanner" /usr/local/lib/libhostscannerimv. so. 0. 8. 4
IMC "UsbScanner" /usr/local/lib/libusbscannerimv. so. 0. 8. 4
IMC "File" /usr/local/lib/libclientcheckerimv. so. 0. 8. 4
```

3　测试与分析

3.1　可信网络连接测试

首先服务器端开启认证服务程序,监听网络接入请求。网络请求接入客户端使用代理软件发起认证,把自身的用户名、密码以及证书等发给认证服务器端,请求接入网络;如果用户名和密码都符合要求,客户端按照服务器端的策略要求收集自身的文件完整性信息,继续发送到认证服务器端;如果完整性信息也符合策略要求,则允许接入网络,否则进入隔离区进行修复,以下步骤为使用插入非法外设导致平台不可信,引起主备切换测试过程:

(1) 首先服务器端开启认证服务,监听连接请求。

(2) 客户端输入用户名密码,执行认证请求命令,认证通过之后,再检测客户端杀毒软件、端口、文件完整性、外设,如果都符合既定策略,则进入可信区。

(3) 插入非法 U 盘进行测试,由于检测到有非法 USB 设备接入,从可信网转入隔离区,要求进行修复。拔掉 U 盘并再次发起可信认证,验证通过后,进入可信区。

实验结果表明,该系统可以根据既定策略准确的对网络接入客户端进行访问控制,防止由于终端计算机不可信而导致网内的其他计算机受到安全危胁。

3.2　冗余切换测试

服务器端冗余测试采用人为强制关闭服务认证程序的方式,然后记录对端冗余机开启服务的时间,计算该时间与关闭服务时的时间间隔。经 50 次测试,测试结果表明网络中断时间不超过 3s。客户端冗余测试,采用插入 U 盘的方式使本机不可信,记录对端冗余客户端启动应用业务的时间,计算该时间与断开网络的时间间隔。经过 50 次的测试,测试结果表明网络中断时间不超过 2s。图 6 展示了其中的 13 次切换时间间隔。

其中服务器端冗余平均切换时间为 2569ms,客户端平均切换时间间隔为 1361ms,服务器端平均切换时间间隔略高于客户端,经进一步测试分析发现,主要是由于服务器端执行 Radius 认证服务程序自身引起的延迟所致。

对于冗余服务器和客户端分别进行主备切换成功率测试,测试结果如表 2 所示。

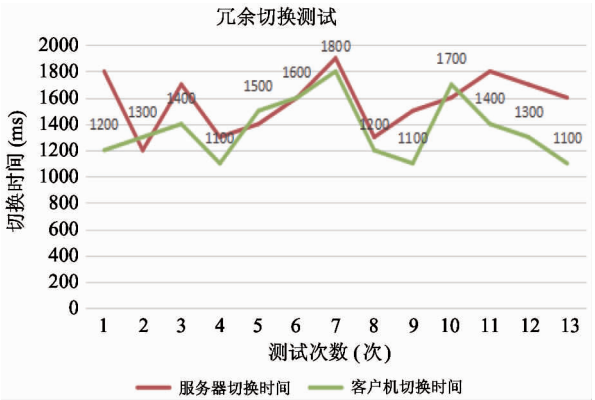

图 6　冗余切换测试图

表 2　冗余测试成功率

测试项	测试次数		成功次数		成功率	
	服务器端	客户端	服务器端	客户端	服务器端	客户端
插入非法 U 盘	20	20	19	20	95%	100%
打开非法网络端口	10	10	10	10	100%	100%
修改关键文件	10	10	10	10	100%	100%
卸载杀毒软件	10	10	10	10	100%	100%
合计	50	50	49	50	98%	100%

从测试结果可以看到失效切换成功率达 98%。经对切换失败的原因分析，主要是操作系统对 USB 设备识别过程有一定延迟，两次实验的时间间隔过短，导致系统首次切换失败。但在系统再次执行切换动作时，可以顺利完成切换，能够满足实际应用需要。

3.3　性能测试

为了检测心跳服务及可信完整性度量引起的系统资源占用情况，针对客户端执行网络接入认证前后的 CPU 占用率和内存使用情况以及服务器端执行服务认证程序前后的 CPU 占用率和内存使用情况进行对比测试，测试结果如图 7 所示。

深色实线和虚线分别是客户端执行网络接入请求前后的 CPU 占用率，两者相差为 2% ~5%。线色实线和虚线分别代表服务器端认证服务开启前后的 CPU 占用情况，CPU 消耗 1.1% ~3.5%。图 8 展示了客户端执行网络接入请求前后的内存使用情况以及服务器端执行服务认证程序前后的内存使用情况。

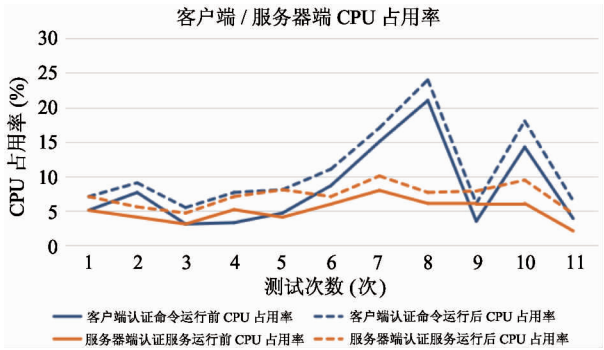

图 7　客户端/服务器端 CPU 占用率

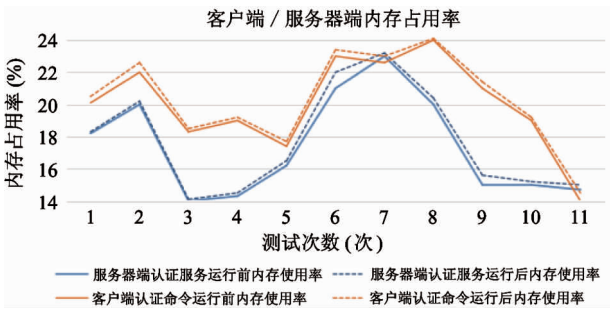

图 8　客户端/服务器端内存占用率

由数据可知，无论是客户端还是服务器端，执行程序前后内存使用相差不超过 1%。综合 CPU 使用率、内存占用率测试结果，客户端和服务器端对系统资源的消耗都在可以接受的范围内。

4　结　论

本文是基于高可用(HA)集群中的双机热备思想，分别对 TNC(可信网络连接)技术中的服务器端和客户端进行了冗余设计并对可信网络协议进行了相应的改造和扩展，在实现网络可信接入的同时，保证了网络资源的不间断使用。测试表明，所提出的方法能够满足高可靠应用环境的使用要求。下一步将在此基础上，针对网络设备的可信接入问题进行深入研究。

参考文献

[1] US Government. Cyberspace Policy Review. http://www.

Whiteho-use. gov/assets/documents/Cyberspace _ Policy _ Review _ final. pdf:USA. gov,2009

[2] 林闯,彭雪梅. 可信网络研究. 计算机学报,2005,28(5):751-757

[3] 张焕国,陈璐,张立强. 可信网络连接研究. 计算机学报,2010, 33(04):706-717

[4] Trusted Computing Group. TCG Specification Trusted Network Connect-TNC Architecture for Interoperability Revision1. 1. http://www. trustedcomputinggroup. org: TCG, 2006

[5] Trusted Computing Group(TCG). TCG Trusted Network Connect TNC Architecture for Interoperability Specification(version 1.0):2005-03

[6] 柳阳. 基于 HA 集群的 Linux 多机互备份系统的研究与应用:[硕士学位论文]. 广州:华南理工大学, 2011

[7] 颜菲,任江春,戴葵等. 基于 TNC 的安全认证协议的设计与实现. 计算机工程, 2007, 33(12):160-162 + 165

[8] Rehbock S, Hunt R. Trustworthy clients:Extending TNC to web-based environments. *Computer Communications*, 2009,32(5):1006-1013

[9] Fu J, Zeng H L. The model of trusted network connected based on creadibility of the hierarchy. In: Proceedings of the Networks Security Wireless Communications and Trusted Computing (NSWCTC), Wuhan, China, 2010,2: 454-457

[10] 肖跃雷. 可信网络连接关键技术研究及其应用:[博士学位论文]. 西安:西安电子科技大学通信工程学院, 2013

[11] 沈昌祥,公备. 基于国产密码体系的可信计算体系框架. 密码学报, 2015,2(5):381-389

[12] 金雷,徐开勇,李剑飞等. 基于 TNC 的网络终端认证模型设计. 计算机应用与软件, 2015,32(12):321-325

[13] 蒋华,张乐乾,阮玲玲. 基于公钥密码体制的 802. 1x 双向认证研究. 计算机应用与软件, 2016,2:290-293

A method for construction of trusted networks suitable for high availability environments

Tan Li*, Zhang Yaming*, Yang Minghua**, Hu Jipeng*

(*School of Computer and Information Engineering, Beijing Technology and Business University, Beijing 100048)

(**The Fourth School of The Rocket Force Equipment Academy, Beijing 100094)

Abstract

To secure the credibility of network access terminals and the network security, a method for trusted network construction suitable for high availability environments was proposed by using the trusted network connection (TNC) technique combined with the high availability (HA) clustering. This method can make a strategy judgement for network access request according to platform ID, version of antivirus software, illegal storage device as well as network ports. At the same time, to the purpose of high availability, servers and clients are all designed on hot-standby. The availability of this method was verified by a test.

Key words: trusted network connect (TNC), high availability (HA), cluster, hot-standby, trusted computing