强机保墙

Mr. Firewall's Talk Show

1-18期合辑

企业网络资料部 防火墙产品部联合出品 2014年第1期总第1期

2014年第1期总第1期 基础知识篇+安全策略篇+攻击防范篇+NAT篇

基础知识

什么是防火墙

安全区域：划地而治 等级森严

状态检测和会话机制

安全策略

安全策略初体验

安全策略发展历程详解

ASPF：隐形通道

攻击防范

单包攻击及防御

流量型攻击之SYN Flood

应用层攻击及防御

NAT

一墙当关，万夫上网—源NAT

NAT Server三十二字真言

双剑合璧，无往不利—双向NAT

HUAWEI

序

话说数通江湖早已被南北两派占据多年，北派Router，以融合天下各家奇招著称，据守网络要冲，上至天庭下达乡野，人脉甚广；南派LAN Switch，专攻以太之术，潜伏乡野，地方势力强大。公元一九八九年，名不见经传的Firewall悄然出世，该派自称“门户卫士”，专守城池门户重地，善筑隧道、精确防守、行事机密，弟子个个身怀绝技，因此能得真传之人甚少。江湖人士对南北派必杀技—MSTP、OSPF、BGP朗朗上口，但对门户卫士的寻常招式Security Policy、IPSec、NAT都知之甚少。这既是其高深之处，也是致命所在。一个门派若不能发扬光大，就必然走向衰落。虽IP技术日新月异，但门户卫士总部—华为安全论坛雄风不振。当此之时，华为防火墙产品线资深技术专家—强叔横空出世，一手携Firewall，一手轻揽Anti-DDOS、NIP、SVN等新秀，如王阳明讲学，传递内功心法；如诸葛武侯亲临，指点排兵布阵。文字风趣幽默绝不照本宣科，核心技术深入挖掘绝不蜻蜓点水，现网场景演绎逻辑严密绝不简单堆砌，人称“强叔侃墙”。短短3月，单单凭借一“侃”字诀，强叔已坐拥粉丝几千，已然重振门派雄风。而今强叔特推出“强叔侃墙”电子期刊，一为答谢广大强粉，二表传道授业解惑之志。望一年之后，华为安全论坛英杰辈出，华为Firewall名满天下。

徐慧洋

目录

Contents

什么是防火墙

2013 年 9 月，华为在首届企业网络大会上发布下一代防火墙 USG6600，标志着华为防火墙又进入一个新的历史发展阶段。

2013 年 12 月，在 Forrester Research 发布的最新关于网络隔离网关报告中，华为下一代防火墙作为唯一的中国厂商，在安全隔离网关特性上的全面性和质量方面，以 95%以上的超高满足度获得了优秀的评价。

时光倒回至 2001 年，华为发布第一款防火墙插卡，白驹过隙，转眼已经十二年。十二年来，互联网可以说经历着不可预测的高速发展阶段，而华为防火墙以及安全系列产品正是在这发展的浪潮中乘风破浪，逐步成长和壮大，以至今天仍然在不断超越。

俺来自华为防火墙研发团队，人称强叔，曾经的小伙，如今的大叔。今天想在这里，结合华为的防火墙及安全系列产品，与大家东侃侃防火墙的发展历史，关键技术与困惑，西扯扯安全技术发展趋势。与交换机、路由器相比，对防火墙熟悉的同学可能相对较少，强叔希望，防火墙作为网络部署中安全防护的第一道防线，深深地存在各位网络工程师们的脑海里~~

为表诚意，下面，啰嗦的强叔就从防火墙一词讲起。

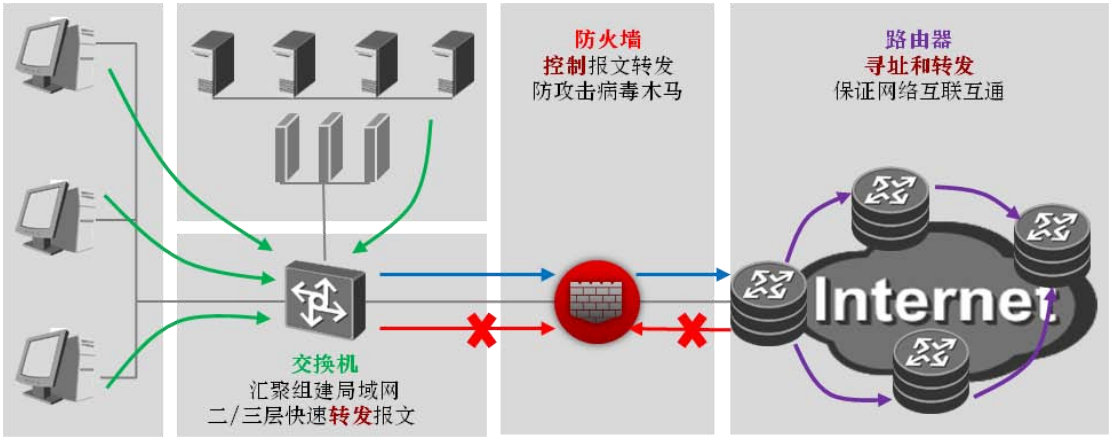
墙，始于防，忠于守。自古至今，墙予人以安全之意。

防火墙，顾名思义，阻挡的是火，此词起源于建筑领域，正是用来隔离火灾，阻止火势从一个区域蔓延到另一个区域。

引入到通信领域，防火墙也正是形象化地体现了这一特点：防火墙这一具体设备，通常用于两个网络之间的隔离。当然，这种隔离是高明的，隔离的是“火”的蔓延，而又保证“人”的穿墙而过。这里的“火”是指网络中的各种攻击，而“人”是指正常的通信报文。

那么，用通信语言来定义，防火墙主要用于保护一个网络区域免受来自另一个网络区域的网络攻击和网络入侵行为。因其隔离、防守的属性，灵活应用于网络边界、子网隔离等位置，具体如企业网络出口、大型网络内部子网隔离、数据中心边界等等。

从上文可以看到，防火墙与路由器、交换机是有区别的。路由器用来连接不同的网络，通过路由协议保证互联互通，确保将报文转发到目的地；交换机则通常用来组建局域网，作为局域网通信的重要枢纽，通过二层/三层交换快速转发报文；而防火墙主要部署在网络边界，对进出网络的访问行为进行控制，安全防护是其核心特性。路由器与交换机的本质是转发，防火墙的本质是控制。



现阶段中低端路由器与防火墙有合一趋势，主要也是因为二者互相功能兼具，变成 all in one 的目标，后文也会讲到华为也发布了一系列低端设备。同时，后文也会对防火墙进行隔离与管控、安全防护的基础和关键技术进行介绍，敬请各位关注。

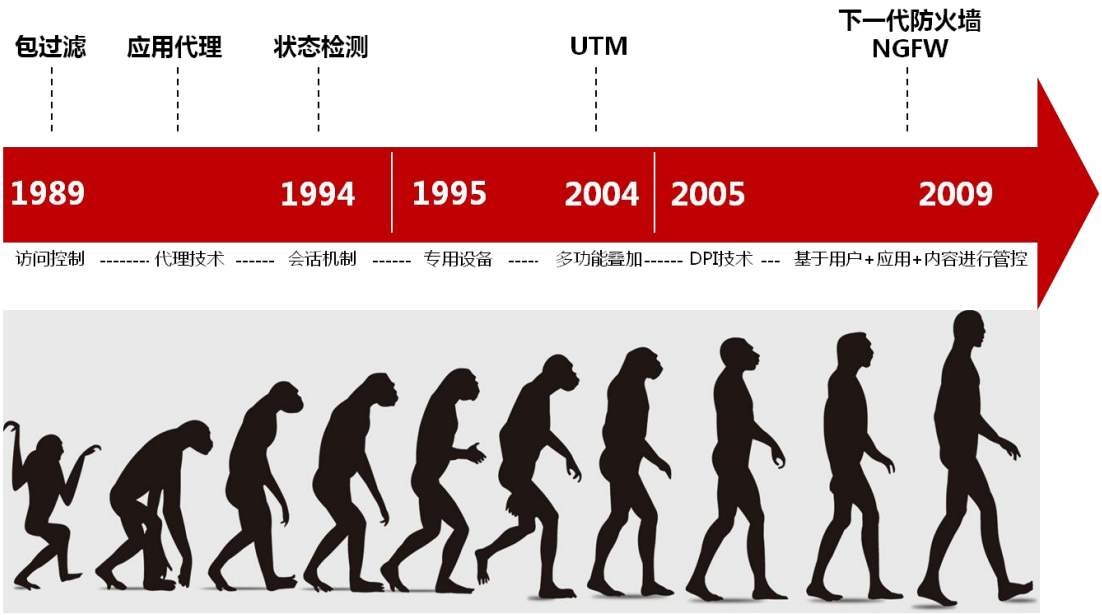
强叔提问

各位小伙伴，你们是怎么与防火墙结缘的呢，你们眼中的防火墙是什么样子，欢迎大家回贴~

防火墙的昨天、今天和明天

各位好，强叔又和大家面见了。上一期为大家介绍了防火墙的基本概念，今天强叔要和大家聊一聊防火墙的昨天、今天和明天，欢迎大家跟着强叔来回顾防火墙的历史，展望防火墙的未来。

与人类的进化史相似，防火墙的发展历史也经历了从低级到高级、从功能简单到功能复杂的过程。在这一过程中，网络技术的不断发展，新需求的不断提出，推动着防火墙向前发展演进。



最早的防火墙可以追溯到上世纪 80 年代末期，距今已有二十多年的历史。在这二十多年间，防火墙的发展过程大致可以划分为下面三个时期：

1989 年至 1994 年

- 1989 年产生了包过滤防火墙，实现简单的访问控制，我们称之为**第一代防火墙**。
- 随后出现了代理防火墙，在应用层代理内部网络和外部网络之间的通信，属于**第二代防火墙**。代理防火墙安全性较高，但处理速度慢，而且对每一种应用开发一个对应的代理服务是很难做到的，因此只能对少量的应用提供代理支持。
- 1994 年 CheckPoint 公司发布了第一台基于状态检测技术的防火墙，通过动态分析报文

的状态来决定对报文采取的动作，不需要为每个应用程序都进行代理，处理速度快而且安全性高。状态检测防火墙被称为**第三代防火墙**。

1995 年至 2004 年

- 在这一时期，状态检测防火墙已经成为趋势。除了访问控制功能之外，防火墙上也开始增加一些其他功能，如 VPN。
- 同时，一些专用设备也在这一时期出现了雏形。例如，专门保护 Web 服务器安全的 WAF（Web Application Firewall，Web 应用防火墙）设备。
- 2004 年业界提出了 UTM（United Threat Management，统一威胁管理）的概念，将传统防火墙、入侵检测、防病毒、URL 过滤、应用程序控制、邮件过滤等功能融合到一台防火墙上，实现全面的安全防护。

2005 年至今

- 2004 年后，UTM 市场得到了快速的发展，UTM 产品如雨后春笋般涌现，但面临新的问题。首先是对应用层信息的检测程度受到限制，举个例子，假设防火墙允许“男人”通过，拒绝“女人”通过，那是否允许来自星星的都教授（外星人）通过呢？此时就需要更高级的检测手段，这使得 DPI（Deep Packet Inspection，深度报文检测）技术得到广泛应用。其次是性能问题，多个功能同时运行，UTM 设备的处理性能将会严重下降。
- 2008 年 Palo Alto Networks 公司发布了下一代防火墙，解决了多个功能同时运行时性能下降的问题。同时，还可以基于用户、应用和内容来进行管控。
- 2009 年 Gartner 对下一代防火墙进行了定义，明确下一代防火墙应具备的功能特性。随后各个安全厂商也推出了各自的下一代防火墙产品，防火墙进入了一个新的时代。

强叔解释

关于Checkpoint：以色列的一家安全厂商，发布了第一台基于状态检测技术的商业化防火墙。

关于Palo Alto Networks：美国的一家安全厂商，率先推出了下一代防火墙，称得上是下一代防火墙的开山鼻祖。

关于Gartner：著名的IT 研究与顾问咨询公司，其研究范围覆盖全部IT 产业，众所周知的魔力象限就出自 Gartner。在 2013 年华为成为首家进入 Gartner 防火墙和UTM 魔力象限的中

国厂商，充分证明了华为安全产品的质量和实力。

从防火墙的发展历史中我们可以看到以下三个最主要的特点：

- 第一点是访问控制越来越精确。从最初的简单访问控制，到基于会话的访问控制，再到下一代防火墙上基于应用、用户和内容来做访问控制，都是为了实现更有效更精确地访问控制。
- 第二点是防护能力越来越强。从早期的隔离功能，到逐渐增加了入侵检测、防病毒、URL过滤、应用程序控制、邮件过滤等功能，防护手段越来越多，防护的范围也越来越广。
- 第三点是性能越来越高。随着网络中业务流量爆炸式增长，对性能的需求也越来越高，各个防火墙厂商通过对硬件和软件架构的不断改进，使防火墙的处理性能与业务流量相匹配。

下一代防火墙并不是终结，网络发展日新月异，新技术、新需求不断涌现，也许用不了几年，防火墙会变得更加高级和智能，也更容易管理和配置，让我们拭目以待。

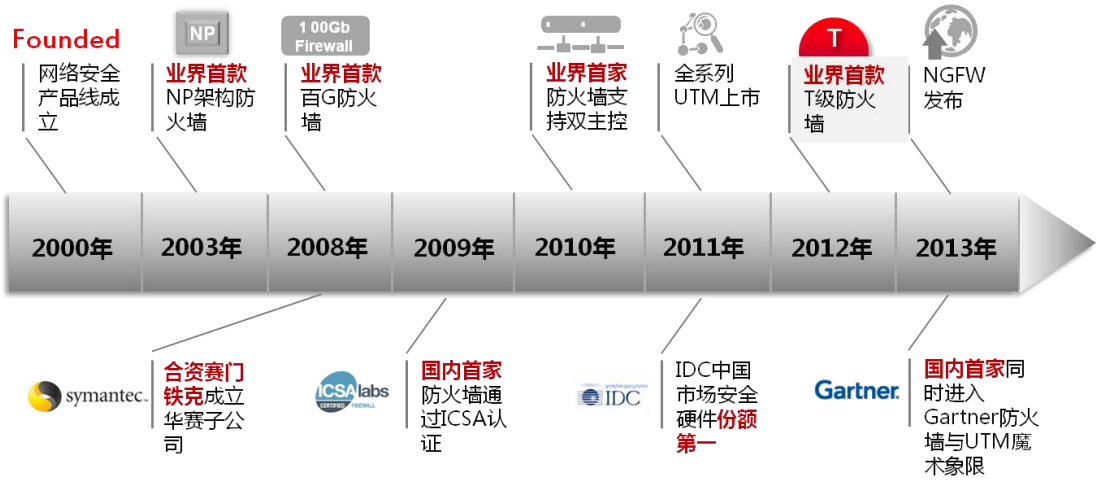
通过上面的内容，大家应该对防火墙发展历史有了初步的认识，接下来强叔会为大家带来华为防火墙产品真容大揭秘，希望大家持续关注“强叔侃墙”系列帖子。

强叔提问

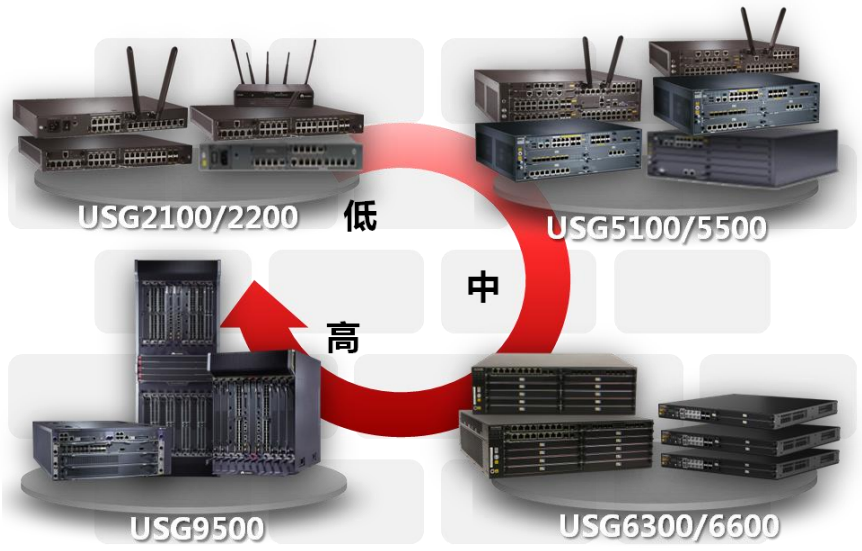
各位小伙伴，未来的防火墙会发展成什么样子呢，或者，还会有防火墙这种设备形态吗，请大家畅所欲言~~

华为防火墙产品一览

在上一篇贴子中，强叔和大家聊了防火墙的前生今世。经过二十多年的发展演变，防火墙的功能越来越强，性能越来越高。同样，华为的防火墙产品也经历了从无到有，逐步壮大的过程。在十几年的时间中，华为防火墙始终坚韧前行，勇于创新，不断突破，取得了一个又一个辉煌。



今天强叔就和大家侃一侃华为的防火墙产品，带着大家纵览华为防火墙全系产品，围观明星产品。耳听为虚眼见为实，先上一张华为防火墙全家福，请大家先睹为快。



华为防火墙产品主要包括 USG2000、USG5000、USG6000 和 USG9500 四大系列，涵盖高、

中、低端设备，型号齐全功能丰富，完全能够满足各种网络环境的需求。

其中，USG2000 和 USG5000 系列定位于 UTM 产品，USG6000 系列属于下一代防火墙产品，USG9500 系列属于高端防火墙产品。下面强叔就挑出几个有代表性的防火墙产品，为大家逐一介绍。

USG2110

首先出场的是小盒子 USG2110，别看它个头小，功能可不差。USG2100 集防火墙、UTM、VPN、路由、无线（WIFI/3G）等十八般武艺于一身，即插即用，配置方便，为客户提供安全、灵活、便捷的一体化组网和接入解决方案。

USG2110 物美价廉，在节省客户投资的同时，能有效降低运维成本，是中小企业、连锁机构、SOHO 企业之必备神器。

USG6600

接下来出场的是超高人气产品 USG6600，它出身于 USG6000 家族，作为华为面向下一代网络环境的防火墙产品，USG6600 提供以应用层威胁防护为核心的下一代网络安全方案，让网络管理员重新掌控网络，看得更清、管得更细、用得更易。

说它人气超高一点也不为过，IT 风云榜 CIO 信赖优秀产品、IT168 年度技术卓越奖、网络世界 NGFW 横向测评多项第一、Forrester“网络隔离网关”报告高度评价等等，USG6600 发布后就获得了各个方面的关注和好评，足以证明它的超高人气。

要说起 USG6600 的优点，那可是滔滔不绝：最精准的应用访问控制、6000+应用识别、多种用户认证技术、全面的未知威胁防护、最简单的安全管理、最高的全业务性能体验。。。。。根本停不下来！在后续的贴子中，强叔会专门推出 USG6600 专题介绍，敬请期待。

USG9500

最后登场的是大块头 USG9500，大块头有大智慧，大体格有大心脏。作为业界首款 T 级数据中心防火墙，USG9500 刚刚成功通过业界权威第三方安全测评机构美国 NSS 实验室的测试，获评为业界最快的防火墙！

USG9500 采用分布式软硬件设计，融合了多种行业领先的专业安全技术，将交换、路由、安全服务整合到统一的设备中，在大型数据中心、大型企业、教育、政府、广电等行业和典型场景得到广泛应用。

USG9520　　USG9560　　USG9580

天下武功，唯快不破。在以高速网络为基石的云计算时代，USG9500 系列作为华为最高端的防火墙产品，如磐石屹立不倒，似大海吸纳百川，从容应对海量访问和数据洪流。至强性能扛万斤重担，应云而生保万全之策，正是 USG9500 系列防火墙产品的真实写照。

结束语

通过上面的介绍，强叔带着大家认识了几款华为防火墙产品，大家看过之后是不是感觉意犹

未尽呢？其实除了这几个产品之外，华为公司还有多款防火墙产品，如需进一步了解华为防火墙产品，请猛戳华为网络安全产品。

强叔提问

各位网络攻城狮，你们调试过个头最小的防火墙有多小，个头最大的防火墙有多大呢，欢迎大家踊跃跟帖~~

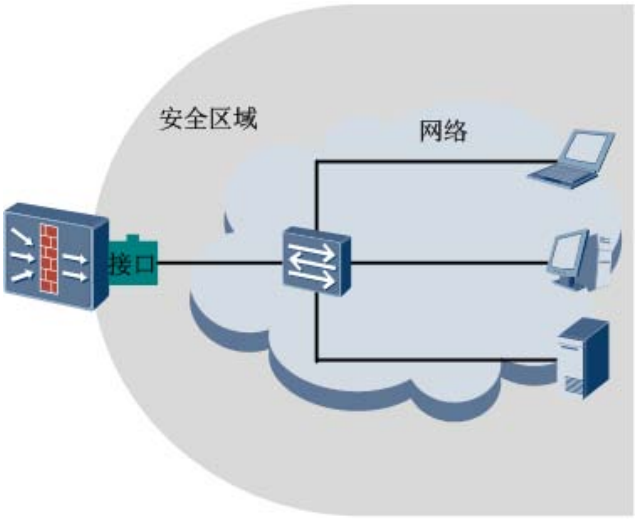
安全区域：划地而治 等级森严

各位好，前几期强叔和大家聊了防火墙的概念和发展历史，并为大家介绍了华为的防火墙产品，相信大家对防火墙已经有了一个初步的认识。从今天开始，强叔将为大家讲解防火墙的技术知识，继续探究防火墙的精彩世界。

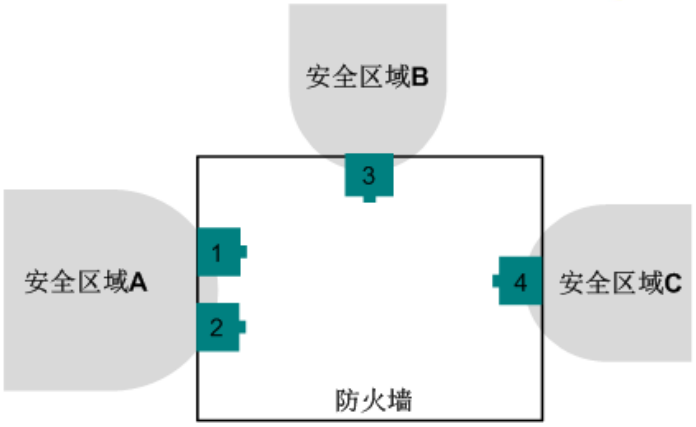
在第一篇贴子中我们提到，防火墙主要部署在网络边界起到隔离的作用，那么在防火墙上如何来区分不同的网络呢？

为此，我们在防火墙上引入了一个重要的概念：安全区域（Security Zone），简称为区域（Zone）。安全区域是一个或多个接口的集合，是防火墙区别于路由器的主要特性。防火墙通过安全区域来划分网络、标识报文流动的“路线”，当报文在不同的安全区域之间流动时，才会触发安全检查[1]。

我们都知道，防火墙通过接口来连接网络，将接口划分到安全区域后，通过接口就把安全区域和网络关联起来。通常说某个安全区域，就可以表示该安全区域中接口所连接的网络。接口、网络和安全区域的关系所下图所示。



通过把接口划分到不同的安全区域中，就可以在防火墙上划分出不同的网络。如下图所示，我们把接口 1 和接口 2 放到安全区域 A 中，接口 3 放到安全区域 B 中，接口 4 放到安全区域 C 中，这样在防火墙上就存在了三个安全区域，对应三个网络。

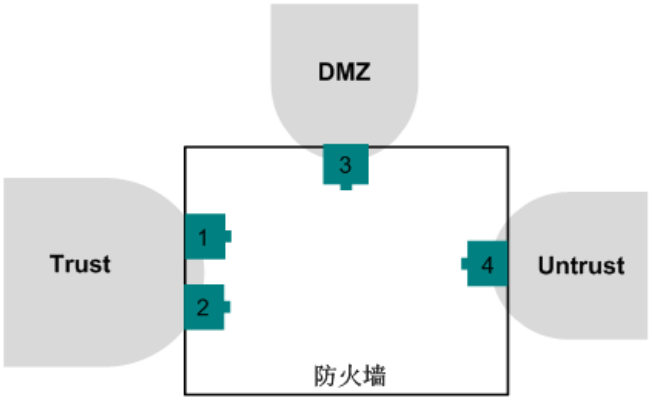


华为防火墙产品上默认已经为大家提供了三个安全区域，分别是 Trust、DMZ 和 Untrust，光从名字看就知道这三个安全区域很有内涵，下面强叔就为大家逐一介绍：

- Trust 区域，该区域内网络的受信任程度高，通常用来定义内部用户所在的网络。
- DMZ 区域[2]，该区域内网络的受信任程度中等，通常用来定义内部服务器所在的网络。
- Untrust 区域，该区域代表的是不受信任的网络，通常用来定义 Internet 等不安全的网络。

在网络数量较少、环境简单的场合中，使用默认提供的安全区域就可以满足划分网络的需求。当然，在网络数量较多的场合，您还可以根据需要创建新的安全区域。

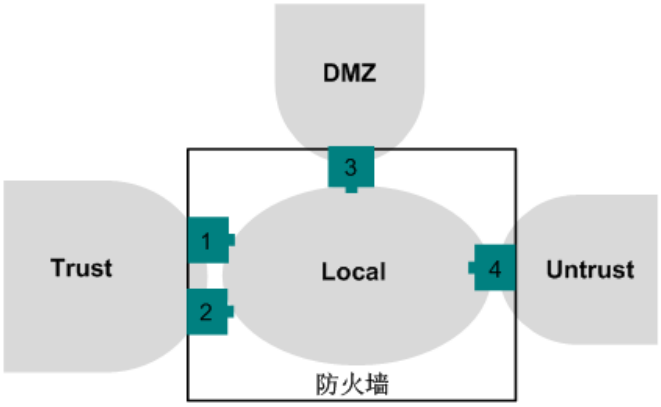
如下图所示，假设接口 1 和接口 2 连接的是内部用户，那我们就把这两个接口划分到 Trust 区域中；接口 3 连接内部服务器，将它划分到 DMZ 区域；接口 4 连接 Internet，将它划分到 Untrust 区域。



由此我们可以得知，不同网络间互访时报文在防火墙上所走的路线。例如，当内部网络中的用户访问 Internet 时，报文在防火墙上的路线是从 Trust 区域到 Untrust 区域；当 Internet 上的用户访问内部服务器时，报文在防火墙上的路线是从 Untrust 区域到 DMZ 区域。

除了在不同网络之间流动的报文之外，还存在从某个网络到达防火墙本身的报文（例如我们登录到防火墙上进行配置），以及从防火墙本身发出的报文，如何在防火墙上标识这类报文的路线呢？

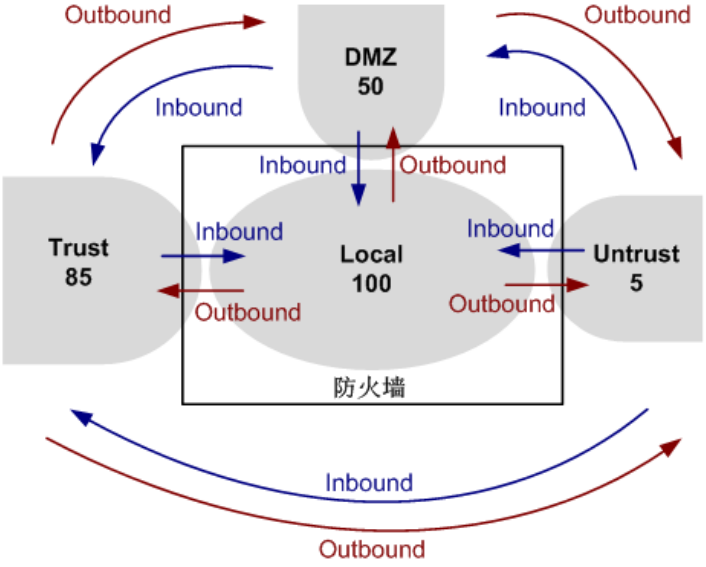
如下图所示，防火墙上提供了 Local 区域，代表防火墙本身。凡是由防火墙主动发出的报文均可认为是从 Local 区域中发出，凡是需要防火墙响应并处理（而不是转发）的报文均可认为是由 Local 区域接收。



关于 Local 区域，强叔还要再提醒一句，Local 区域中不能添加任何接口，但防火墙上所有接口本身都隐含属于 Local 区域。也就是说，报文通过接口去往某个网络时，目的安全区域是该接口所在的安全区域；报文通过接口到达防火墙本身时，目的安全区域是 Local 区域。

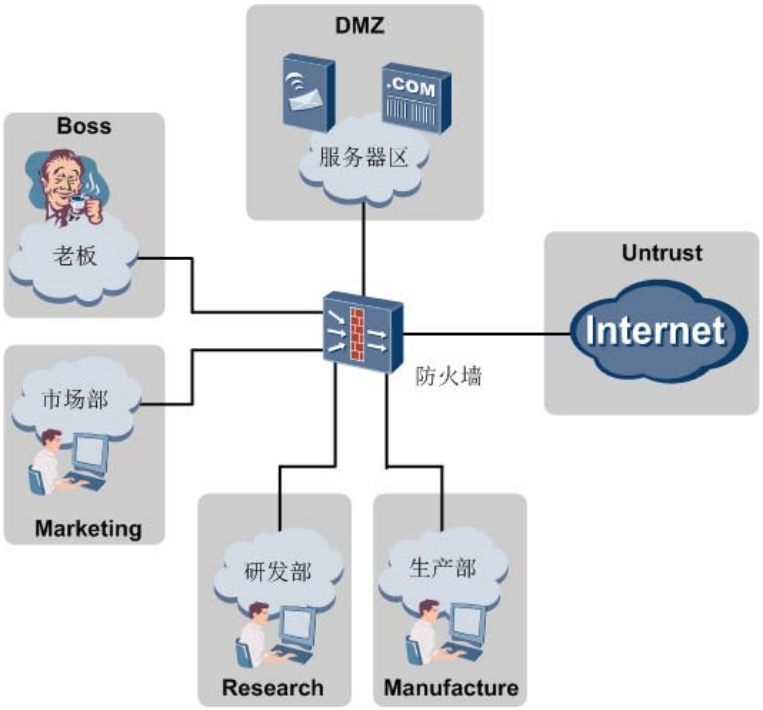
现在我们就可以把经过防火墙的流量和防火墙本身的流量都标识出来了，前面介绍过，不同的网络受信任的程度不同，在防火墙上用安全区域来表示网络后，怎么来判断一个安全区域的受信任程度呢？在华为防火墙上，每个安全区域都有一个唯一的安全级别，用 1～100 的数字表示，数字越大，则代表该区域内的网络越可信。对于默认的安全区域，它们的安全级别是固定的：Local 区域的安全级别是 100，Trust 区域的安全级别是 85，DMZ 区域的安全级别是 50，Untrust 区域的安全级别是 5。

级别确定之后，安全区域就被分成了三六九等，高低有别。报文在两个安全区域之间流动时，我们规定：**报文从低级别的安全区域向高级别的安全区域流动时为入方向（Inbound），报文从由高级别的安全区域向低级别的安全区域流动时为出方向（Outbound）。**报文在两个方向上流动时，将会触发不同的安全检查。下图标明了 Local 区域、Trust 区域、DMZ 区域和 Untrust 区域间的方向。



通过安全区域，防火墙上划分出了等级森严、关系明确的网络，防火墙成为连接各个网络的节点。以此为基础，防火墙就可以对各个网络之间流动的报文进行安全检查和实施管控策略。

下面给出了防火墙部署在企业内部的真实环境组网图。从图中我们可以看出，企业内部网络中的用户、服务器，以及位于外部的 Internet，都被划分到不同的安全区域中了，防火墙对各个安全区域之间流动的报文进行安全检查。



上面我们花了很大的篇幅来介绍安全区域，主要目的还是要说明安全区域的重要性。希望通过强叔的介绍，可以让大家了解安全区域的作用，掌握安全区域之间的关系，为后面进一步

学习防火墙知识打好基础。

强叔解释

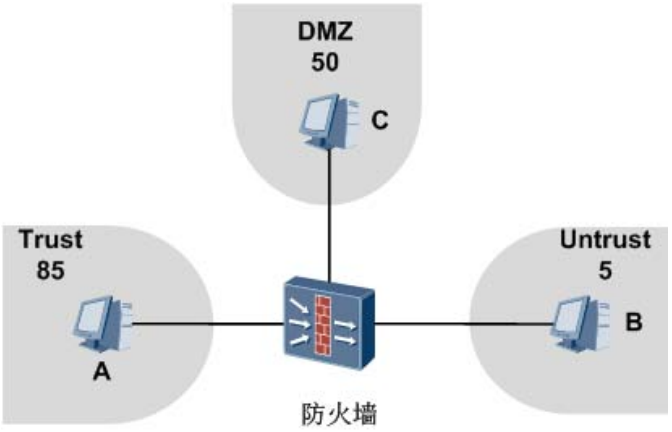
1：默认情况下，报文在不同的安全区域之间流动时，才会触发安全检查，在同一个安全区域中流动时，不会触发安全检查。同时，华为的防火墙也支持对同一个安全区域内经过防火墙的流量进行安全检查，更加灵活实用。

2：DMZ（Demilitarized Zone）起源于军方，是介于严格的军事管制区和松散的公共区域之间的一种部分管制的区域。防火墙引用了这一术语，指代一个与内部网络和外部网络分离的安全区域。

强叔提问

如下图所示，安全区域的名称和级别都已经注明，请问 A、B、C 之间互访时报文流动的路线（包括方向）是什么样的呢？

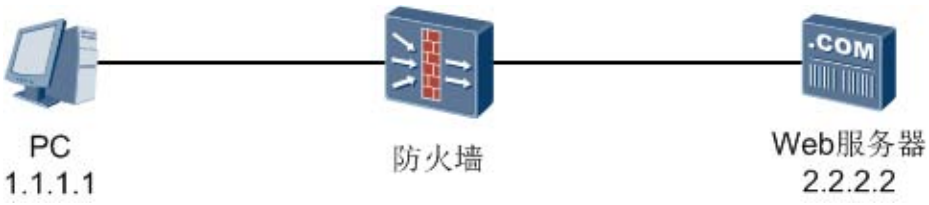
强叔先给出一个例子：A 访问 B 时的路线是 Trust 区域到 Untrust 区域的 Outbound 方向。



状态检测和会话机制

论坛的各位小伙伴，大家是否还记得，在第二篇介绍防火墙发展历史的贴子中，我们提到了第三代防火墙，也就是状态检测防火墙。状态检测防火墙的出现是防火墙发展历史上里程碑式的事件，而其所使用的状态检测和会话机制，目前已经成为防火墙产品的基本功能，也是防火墙实现安全防护的基础技术。今天，强叔就和大家来聊一聊状态检测和会话。

首先，我们从状态检测防火墙产生的背景说起。请大家先看一个简单的网络环境，如下图所示，PC 和 Web 服务器位于不同的网络，分别与防火墙相连，PC 与 Web 服务器之间的通信受到防火墙的控制。



当 PC 需要访问 Web 服务器浏览网页时，在防火墙上必须配置如下的一条规则，允许 PC 访问 Web 服务器的报文通过。

编号	源地址	源端口	目的地址	目的端口	动作
1	1.1.1.1	*	2.2.2.2	80	允许通过

在这条规则中，源端口处的*表示任意的端口，这是因为 PC 在访问 Web 服务器时，它的操作系统决定了所使用的源端口，例如，对于 WINDOWS 操作系统来说，这个值可能是 1024~65535 范围内任意的一个端口。这个值是不确定的，所以这里设定为任意端口。

配置了这条规则后，PC 发出的报文就可以顺利通过防火墙，到达 Web 服务器。然后 Web 服务器将会向 PC 发送回应报文，这个报文也要穿过防火墙才能到达 PC。在状态检测防火墙出现之前，包过滤防火墙还必须配置如下所示的规则 2，允许反方向的报文通过。

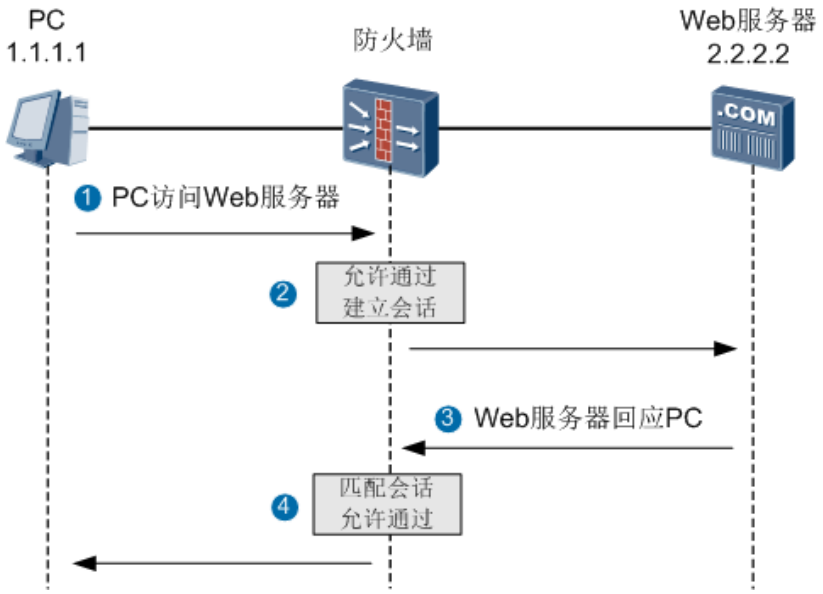
编号	源地址	源端口	目的地址	目的端口	动作
1	1.1.1.1	*	2.2.2.2	80	允许通过
2	2.2.2.2	80	1.1.1.1	*	允许通过

在规则 2 中，目的端口也设定为任意端口，因为我们无法确定 PC 访问 Web 服务器时使用的源端口，要想使 Web 服务器回应的报文都能顺利穿过防火墙到达 PC，只能将规则 2 中的目的端口设定为任意端口。

如果 PC 位于受保护的网络中，这样处理将会带来很大的安全问题。规则 2 将去往 PC 的目的端口全部开放，外部的恶意攻击者伪装成 Web 服务器，就可以畅通无阻地穿过防火墙，PC 将会面临严重的安全风险。

接下来让我们看一下状态检测防火墙怎么解决这个问题。还是以上面的网络环境为例，首先我们还是需要在防火墙上设定规则 1，允许 PC 访问 Web 服务器的报文通过。当报文到达防火墙后，防火墙允许报文通过，同时还会针对 PC 访问 Web 服务器的这个行为建立会话（Session），会话中包含了 PC 发出的报文信息，如地址和端口等。

当 Web 服务器回应给 PC 的报文到达防火墙后，防火墙会把报文中的信息与会话中的信息进行比对，发现报文中的信息与会话中的信息相匹配，并且符合协议规范对后续包的定义，则认为这个报文属于 PC 访问 Web 服务器行为的后续回应报文，直接允许这个报文通过，如下图所示。



而恶意攻击者即使伪装成 Web 服务器向 PC 发起访问，由于这类报文不属于 PC 访问 Web 服务器行为的后续回应报文，防火墙就不会允许这些报文通过。这样就解决了包过滤防火墙大范围开放端口带来的安全风险，同时也保证了 PC 可以正常访问 Web 服务器。

总结一下，包过滤防火墙只根据设定好的静态规则来判断是否允许报文通过，它认为报文都

是无状态的孤立个体，不关注报文产生的前因后果。而状态检测防火墙的出现正好弥补了包过滤防火墙的这个缺陷，状态检测防火墙使用基于连接状态的检测机制，将通信双方之间交互的属于同一连接的所有报文都作为整体的数据流来对待。在状态检测防火墙看来，同一个数据流内的报文不再是孤立的个体，而是存在联系的。为数据流的第一个报文建立会话，数据流内的后续报文直接根据会话进行转发，提高了转发效率。

接着我们就来进一步了解一下会话，会话是通信双方的连接在防火墙上的具体体现，代表两者的连接状态，一条会话就表示通信双方的一个连接。防火墙上多条会话的集合就叫做会话表（Session table），先看一个标准的会话表项：

```
http  VPN:public --> public 1.1.1.1:2049-->2.2.2.2:80
```

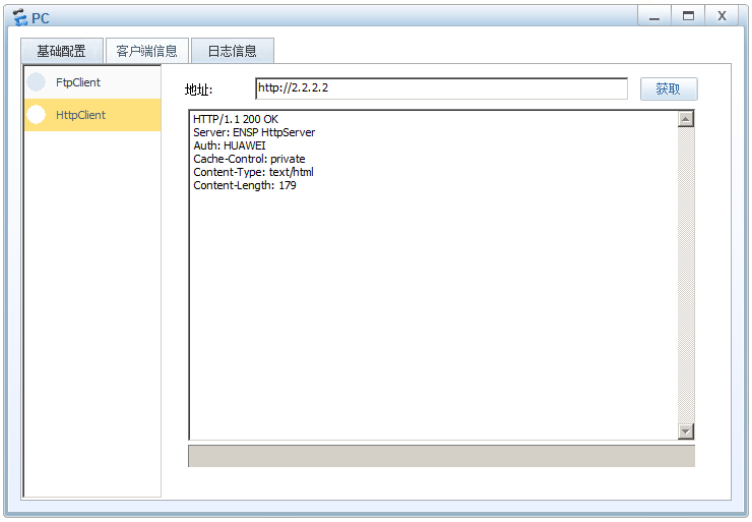
我们重点介绍这个表项中的关键字段：http 表示协议（此处显示的是应用层协议，该会话在 IP 报文头中的协议是 TCP 协议），1.1.1.1 表示源地址，2049 表示源端口，2.2.2.2 表示目的地址，80 表示目的端口。我们是如何区分源和目的呢？其实通过“-->”符号就可以直观区分，符号前面的是源，符号后面的是目的。

源地址、源端口、目的地址、目的端口和协议这五个元素是会话的重要信息，我们将这五个元素称之为“五元组”。只要这五个元素相同的报文即可认为属于同一条流，在防火墙上通过这五个元素就可以唯一确定一条连接。

需要注意的是，会话是动态生成的，但不是永远存在的。如果长时间没有报文匹配，则说明通信双方已经断开了连接，不再需要该条会话了。此时，为了节约系统资源，防火墙会在一段时间后删除会话，该时间称为会话的老化时间。

光说不练假把式，下面强叔就使用 eNSP 模拟器来搭建一个简单的网络环境，验证防火墙上的状态检测机制。网络拓扑如下：

防火墙上只配置了一条规则：允许 PC 访问 Web 服务器的报文通过。在 PC 上使用 HttpClient 程序访问 Web 服务器，发现可以成功访问：



在防火墙上使用 **display firewall session table** 命令查看会话表的信息，发现已经建立一条会话：

```
<SRG>display firewall session table
12:17:18  2014/03/26
 Current Total Sessions : 1
  http  VPN:public --> public 1.1.1.1:2054-->2.2.2.2:80
```

说明状态检测机制工作正常，防火墙收到 Web 服务器返回给 PC 的报文后，发现该报文可以匹配到该条会话，即使没有配置允许反方向报文通过的规则，防火墙也允许其通过。

最后，希望通过强叔的介绍，大家可以了解状态检测和会话机制，也希望大家要理论结合实际，多多使用 eNSP 模拟器动手配置。

强叔提问

给出如下一条会话，大家能指出里面的五元组信息吗？

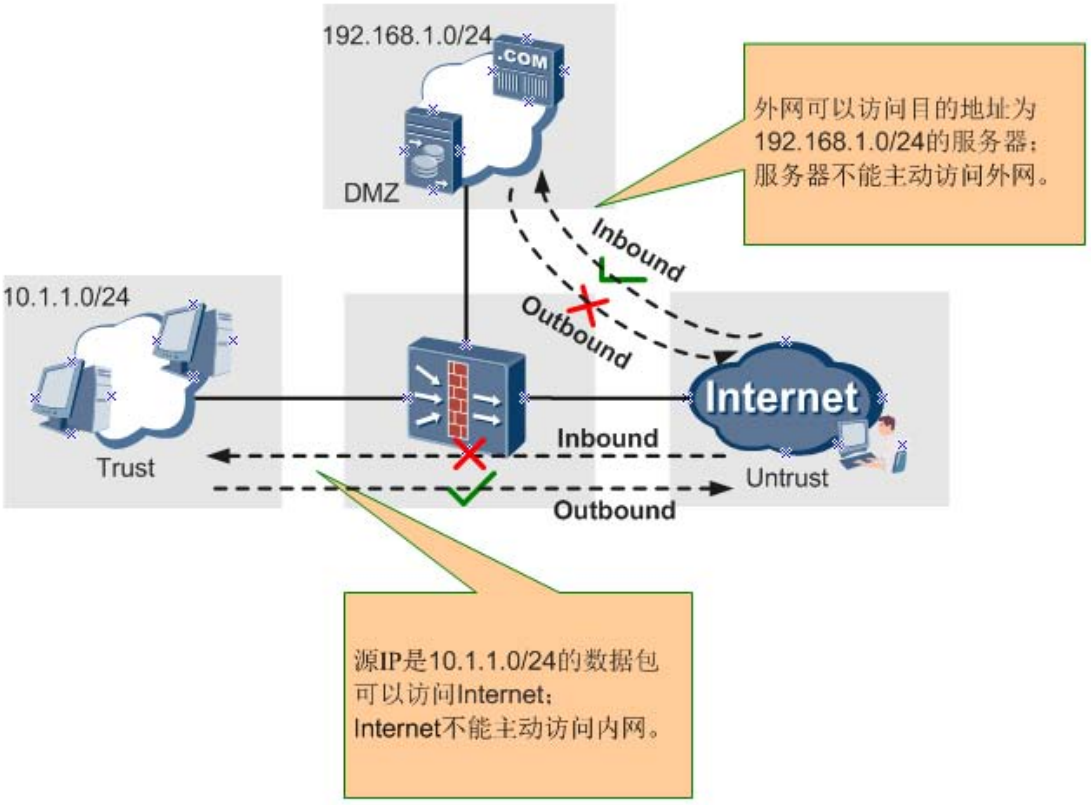
```
telnet    VPN:public --> public 172.16.0.2:51870-->128.18.0.2:23
```

安全策略初体验

大家好，强叔又和你们见面了。通过前几期的介绍，相信大家对防火墙的基础知识已经有了一定的了解。本期开始我们更进一步，迈入“安全策略”篇。

前几期的帖子中，强叔多次提到了“网络隔离”、“访问控制”、“安全检查”等字眼，“安全策略”就是实施这些安全控制的“安检员”哦，他的作用不容小觑！本期先带大家简单了解安全策略的基本概念及发展历程，后续将详细介绍安全策略的内容。

防火墙的基本作用是保护特定网络免受“不信任”的网络的攻击，但是同时还必须允许两个网络之间可以进行合法的通信。安全策略的作用就是对通过防火墙的数据流进行检验，符合安全策略的合法数据流才能通过防火墙。



如上图所示，可以在不同的域间方向应用不同的安全策略进行不同的控制。

安全策略是由匹配条件和动作（允许/拒绝）组成的控制规则，可以基于 IP、端口、协议等属性进行细化的控制。例如下边这条安全策略控制源 IP 是 1.1.1.1 的流量可以访问目的地址为 2.2.2.2 的 Web 服务器。

源地址	源端口	目的地址	目的端口	动作
1.1.1.1	*	2.2.2.2	80	permit

缺省情况下，所有域间的所有方向都禁止报文通过[1]，可以根据需求配置允许哪些数据流通过防火墙的安全策略。

强叔解释

1：对于路由、ARP 等底层协议一般是不受安全策略控制的，直接允许通过。当然这和具体产品实现有关，产品间可能有差异。

另外再啰嗦一下，除了经过防火墙转发的数据流，设备本身与外界互访的数据流也同样受安全策略的控制哦！例如当登录设备、网管与设备对接时，需要配置登录 PC/网管所在安全区域与 Local 域之间的安全策略。

同时为了灵活应对各种组网情况，华为防火墙还支持配置域内策略，也就对同一个安全区域内经过防火墙的流量进行安全检查。当然缺省情况下是允许所有域内报文通过防火墙的。

有人可能要问了，缺省域间不允许数据流通过，我要逐条为允许通过的数据流配置安全策略也太繁琐了？比如我只想控制少数 IP 发起的流量不能经过防火墙，其他 IP 的流量都可以经过防火墙，有什么好办法吗？

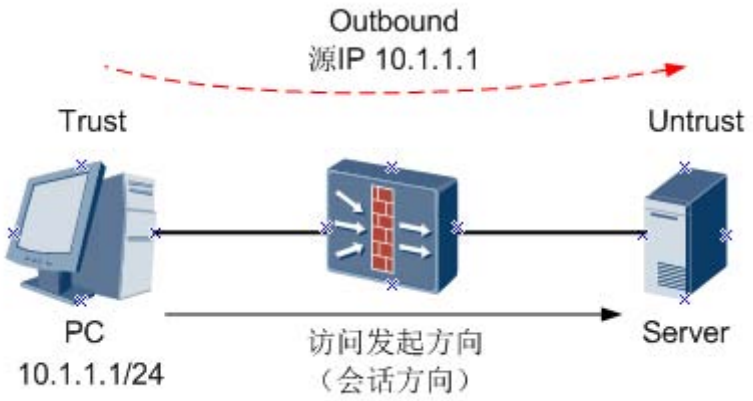
当然有，防火墙出于安全考虑缺省情况拒绝所有流量经过，但是这个缺省情况是可以修改的，这就是“缺省包过滤”。如果流量没有匹配到任何安全策略，将按缺省包过滤的动作进行处理。因此实现上述需求可只配置拒绝流量通过的安全策略，缺省包过滤改为 permit。

类型	源地址	源端口	目的地址	目的端口	动作
安全策略	1.1.1.1 2.2.2.2	*	3.3.3.3	80	deny
缺省包过滤	不涉及				**permit**

安全策略的应用方向

既然一个域间有 Inbound 和 Outbound 两个方向，那是否需要为访问的双向流量同时配置安全策略呢？No，对于同一条数据流，在访问发起的方向上应用安全策略即可，反向报文不需要额外的策略。这点和路由器、交换机包过滤不一样，主要原因就是防火墙是状态检测设备，

对于同一条数据流只有首包匹配安全策略并建立会话，后续包都匹配会话转发。



如上图所示，Trust 域的 PC 访问 Untrust 域的 Server，只需要在 Trust 到 Untrust 的 Outbound 方向上应用安全策略允许 PC 访问 Server 即可，对于 Server 回应 PC 的应答报文会命中首包建立的会话而允许通过。

如果确实需要开放 Server 主动访问 PC 的权限，这时才需要在 Inbound 方向上也应用安全策略。

安全策略的匹配

防火墙将流量的属性与安全策略的条件进行匹配。如果所有条件都匹配，则此流量成功匹配安全策略。如果其中有一个条件不匹配，则未匹配安全策略。

同一域间或域内应用多条安全策略，策略的优先级按照配置顺序进行排列，越先配置的策略优先级越高，越先匹配报文。如果报文匹配到一条策略就不再继续匹配剩下的策略，如果没有匹配到任何策略就按缺省包过滤处理。所以配置策略还是有一定讲究的，要先细后粗。

举个具体的例子：企业 FTP 服务器地址为 10.1.1.1，办公区 IP 段为 10.2.1.0/24，要求禁止两台临时办公 PC（10.2.1.1、10.2.1.2）访问 FTP 服务器。如下这样配置有什么问题？

源地址	源端口	目的地址	目的端口	动作
10.2.1.0/24	*	10.1.1.1	21	permit
10.2.1.1 10.2.1.2	*	10.1.1.1	21	deny

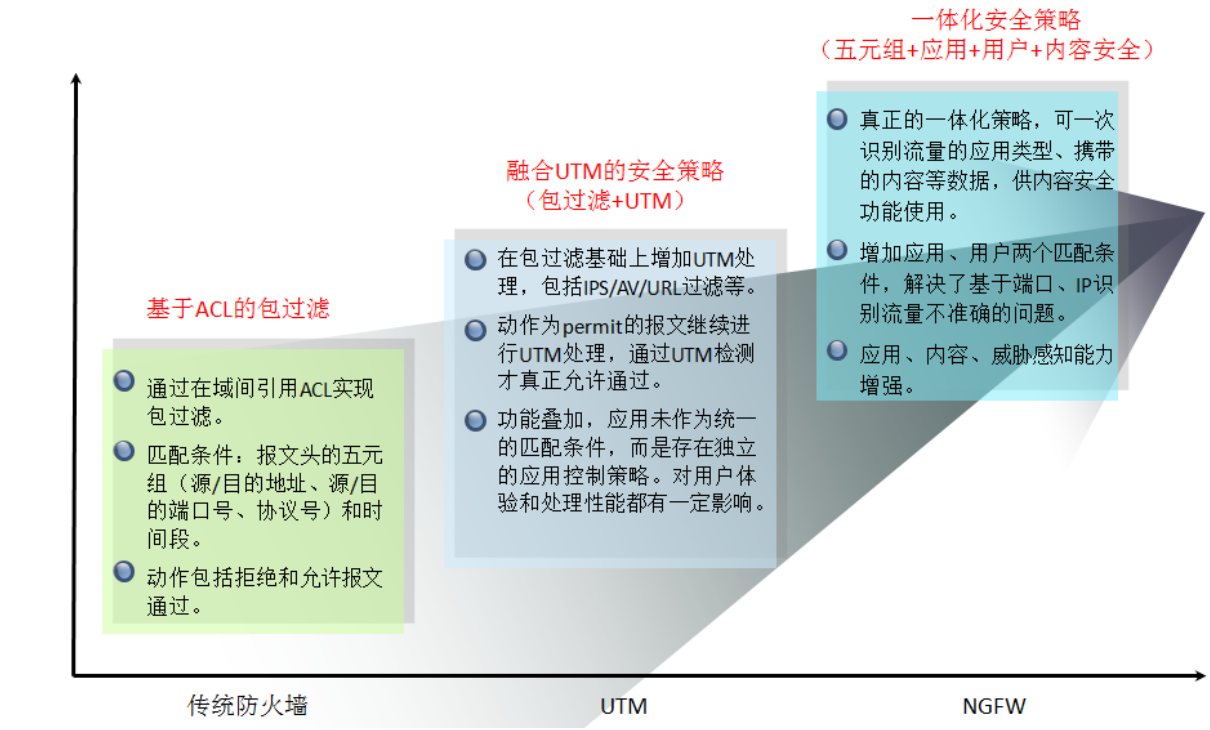
这样配置将无法实现“禁止两台临时办公 PC（10.2.1.1、10.2.1.2）访问 FTP 服务器”的需求，因为这两个 IP 已经命中了第一条宽泛的策略，无法再命中第二条策略。所以两条策略需

要调换顺序。

安全策略发展史

有同学可能要问了，啥安全策略，不就是包过滤吗？那你可 out 了，随着防火墙产品的推陈出新，包过滤也逐渐进化，已经发展成为可以做深度内容检查的“安全策略”。策略匹配条件也已经在“五元组”的基础上增加了用户、应用等匹配条件，还增加了内容安全检测处理。

下图展现了安全策略的发展历程，大家可以看到 NGFW 中已经实现了基于“七元组”的安全策略。细粒度的安全管控使藏匿于流量中的危险分子无所遁形。



强叔解释

“华为防火墙产品一览”那期帖子中已经提到了 USG2000/5000 是 UTM 产品，USG6000 是下一代防火墙 NGFW 产品。USG9500 高端墙也在逐步切换到安全策略，单纯基于 ACL 的包过滤正在逐步退出历史舞台。

好了，本期先开个头，下期强叔将详细介绍每个阶段的安全策略，敬请期待～

顺祝论坛的小伙伴们周末愉快哈～～

强叔提问

Trust 区域的 10.1.1.0/24 这个网段中，除了 10.1.1.2 这个 IP 地址不能访问 Untrust 区域的 Internet 外，其他的 IP 地址都可以访问 Internet。

实现如上需求需要在哪个域间、哪个方向、配置几条安全策略呢？

安全策略发展历程详解

经过上期对安全策略的介绍，我想大家已经了解到安全策略是防火墙中必不可少的基本功能，本期强叔带大家详细了解安全策略的发展历程。

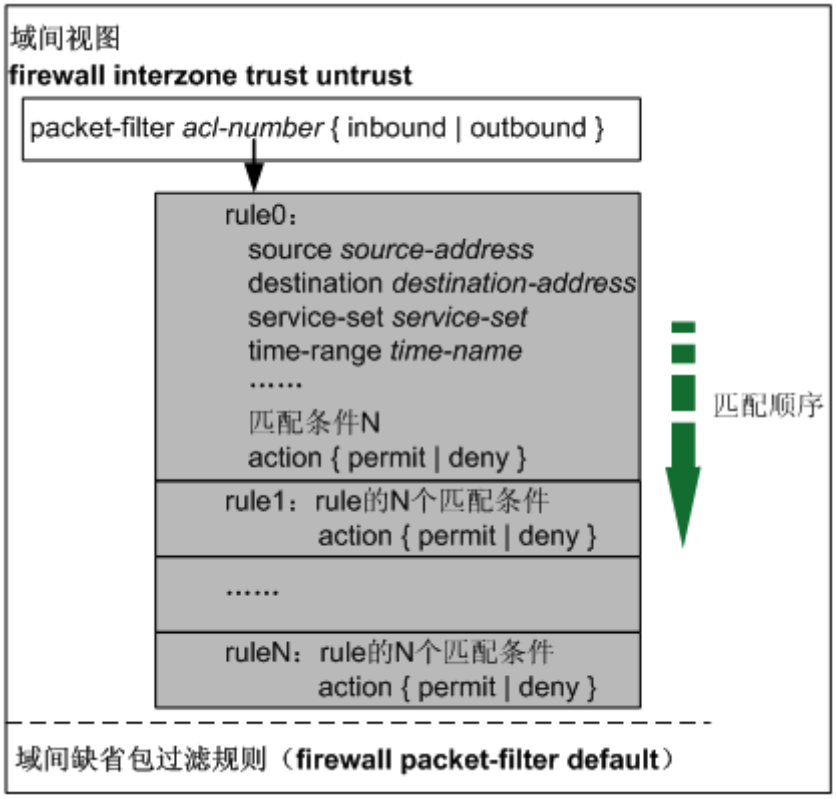
基于 ACL 的包过滤

前期已经提到这种单纯的包过滤正在逐步退出历史舞台，这里只简单介绍一下。

包过滤的处理过程是先获取需要转发数据包的报文头信息，然后和设定的 ACL 规则进行比较，根据比较的结果对数据包进行转发或者丢弃。实现包过滤的核心技术是访问控制列表 ACL。

因此包过滤只能基于 IP 地址、端口号等控制流量是否可以通过防火墙，无法准确识别应用。

基于 ACL 的包过滤的配置方式是先配置好包含多条数据流规则（rule）的 ACL，其中每条 rule 包含数据流的匹配条件和 permit/deny 动作，然后 ACL 再被域间包过滤引用。一个域间只能引用一个 ACL。



发展中期的 UTM 设备安全策略

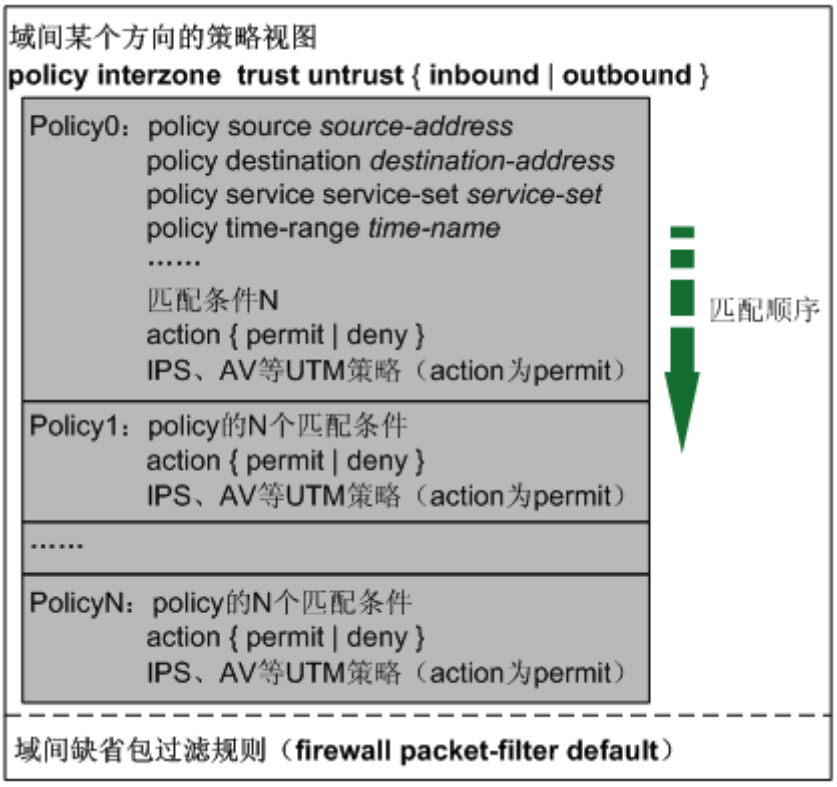
随着 USG2000/5000 系列 UTM 产品的推出，“安全策略”这个概念被提出。之所以不叫包过滤了，是因为策略中集成了 UTM 检测功能，配置方式也由 ACL 方式变为 Policy 方式。

通过下边这个界面可以直观的看到安全策略的组成：包过滤+UTM。不启用 UTM 功能时就是原始的包过滤；动作为 permit 的安全策略可以引用 IPS、AV 等 UTM 策略，对流量进一步进行 UTM 检测，通过检测的流量才能真正通过防火墙。

强叔解释

华为 UTM 产品中已经增加“用户”这个匹配条件，后续在 NGFW 安全策略中再统一介绍。

另外配置方式上也变为 Policy 方式，即在配置安全策略时直接指定匹配数据流的多种条件以及动作，配置更简单。



此时的安全策略已经有一体化安全处理的雏形了，将防火墙包过滤功能和内容安全功能进行了融合，但是还有一定局限性。

了解 UTM 的同学们应该知道，UTM 更多的是体现功能集成，将传统防火墙、入侵防御设备、反病毒设备等集成到一个硬件。UTM 设备的多个安全功能之间的紧密度不高，报文匹配安全策略的匹配条件后需要逐一进入各个 UTM 模块进行检测和处理，如果同时开启多个安全功能，设备性能往往大幅下降。

另外基于应用的管控需要配置额外的应用控制策略，不能直接将应用类型作为策略的匹配条件。例如需要禁止员工使用 IM 应用，此时要额外配置禁止 IM 应用的“应用控制策略”然后再在安全策略引用生效。也就是应用识别与管控需要进入另外一个模块处理。

NGFW 的一体化安全策略

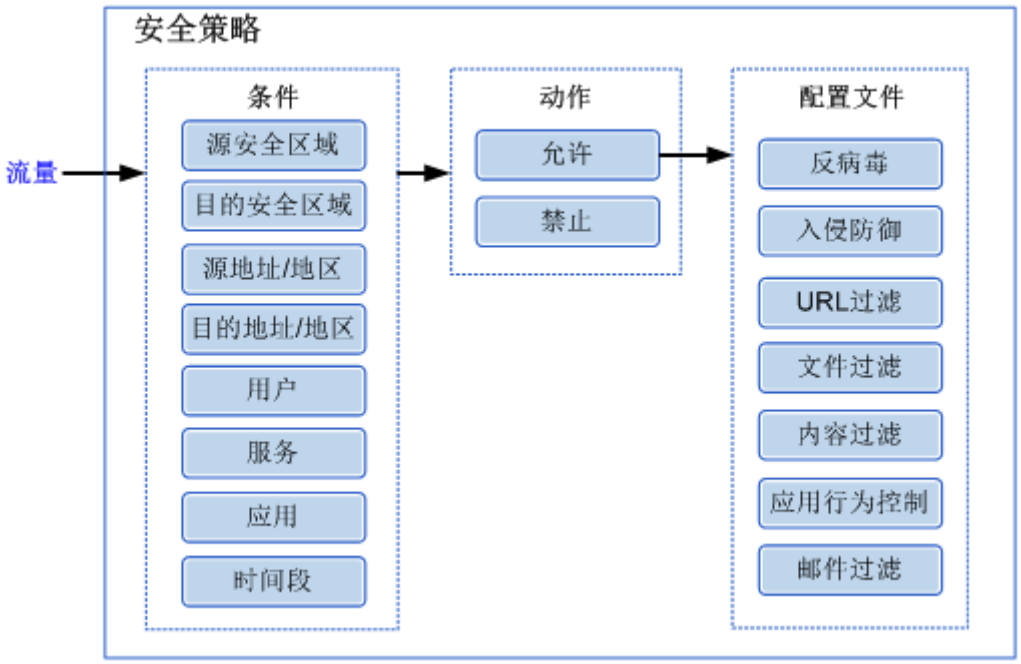
到了 NGFW 阶段，对一体化、应用识别与管控、高性能等要求更高。安全策略充分体现了这些特质，通过应用、用户、内容、威胁等多个维度的识别将模糊的网络环境映射为实际的业

务环境，从而实现精准的访问控制和安全检测。

闲言少叙，说点干货吧，NGFW 安全策略在配置和实现上到底有哪些不同呢？

- NGFW 之前的安全策略都是应用在域间的（安全区域必配），**NGFW 的安全策略应用在全局，安全区域与 IP 地址等一样只是作为可选的匹配条件**。而且安全区域支持多选。

  这样配置更灵活，可以不关注安全区域、域间方向，只需关注访问的源/目的。另外还可以实现跨多域的访问的一次性配置。



```
[USG6600]security-policy
[USG6600-policy-security]rule name abc
[USG6600-policy-security-rule-abc]source-zone trust
[USG6600-policy-security-rule-abc]destination-zone untrust
[USG6600-policy-security-rule-abc]source-address 10.1.1.0 24
[USG6600-policy-security-rule-abc]application app QQLive
[USG6600-policy-security-rule-abc]profile ips default
[USG6600-policy-security-rule-abc]action permit
```

- 缺省包过滤也是全局只有一条，不再区分域间。
- 增加应用作为匹配条件，对应用的阻断/允许直接在安全策略中配置。
- 增加用户作为匹配条件,实现基于用户的管控,即使 IP 发生变化用户的权限也是不变的。
- 也就是应用和用户的识别都融入了防火墙安全策略的处理流程中，无论从配置还是设备处理上都体现了“一体化”。
- 通过高性能的一体化检测引擎实现一次扫描、实时检测，即使开启所有内容安全功能，也不会造成设备性能的大幅下降。

强叔后续将开辟专门的帖子对 NGFW 进行具体介绍，详细的处理过程本帖不过多提及了。

展望未来

安全策略在一体化、智能化、多维度管控的道路上还在继续前行，基于地理位置的控制、根据实际网络流量类型自动调整策略配置的智能策略都将陆续推出。

各位网管和工程师们，期待着只需喝茶看报，防火墙自动进行安全防护的那一天吧～～

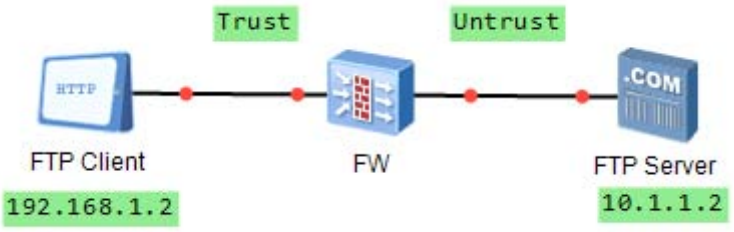
强叔提问

您接触的防火墙的安全策略属于第几阶段？您最想了解什么？都可以留言哦，强叔将会一一解答。

ASPF：隐形通道

大家好，强叔又来了！经过前两期的介绍大家对安全策略应该比较了解了，本期强叔要给大家带来一个神秘人物。

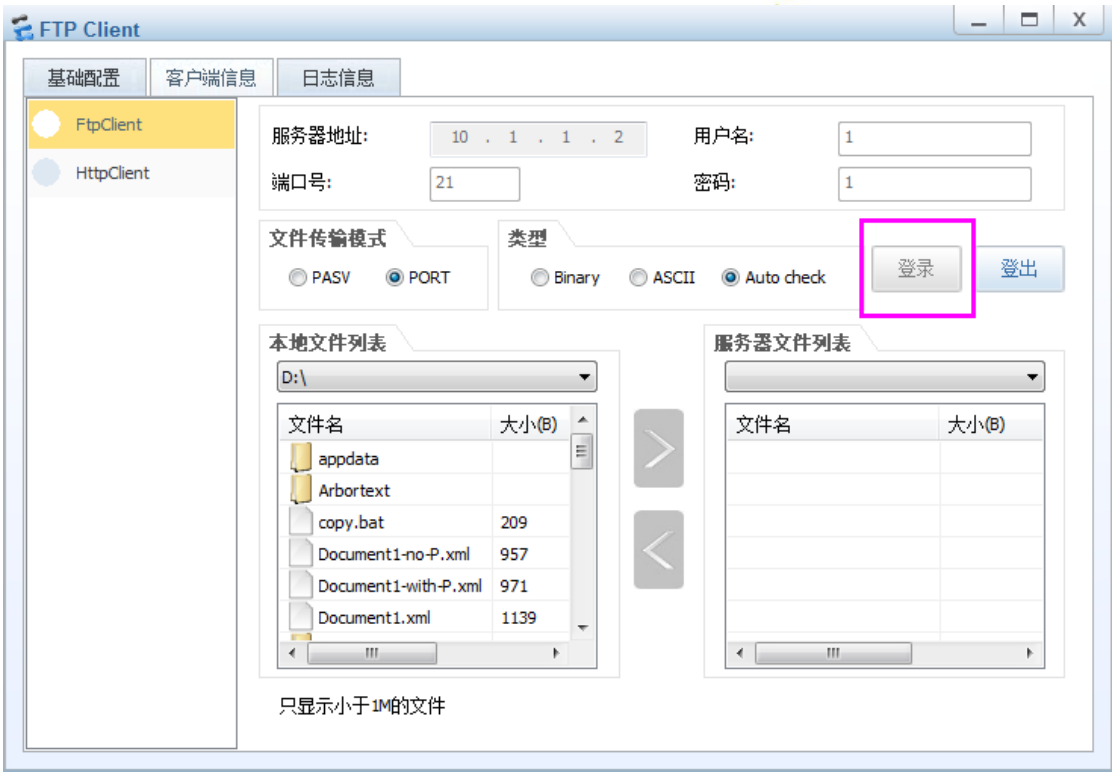
在请出神秘人物之前，我们先来使用 eNSP 实战一把安全策略的配置。通过 eNSP 搭建如下环境，FTP 客户端经过防火墙访问 FTP 服务器，安全策略怎么配置呢？

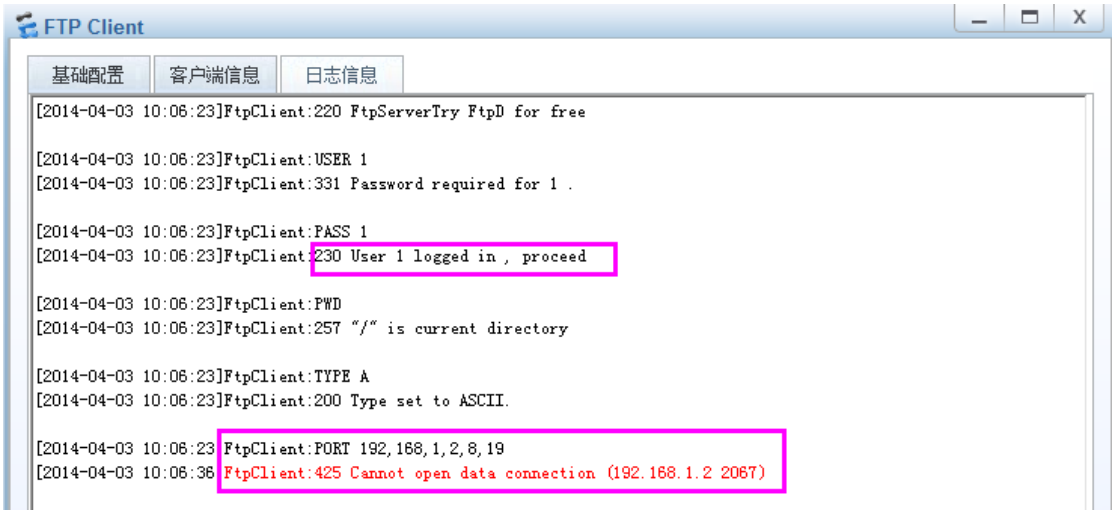


这还不容易，在 Trust 到 Untrust 配置安全策略，指定源/目的 IP、指定协议为 FTP 不就好了。

```
[SRG]policy interzone trust untrust outbound
09:32:43  2014/04/03
[SRG-policy-interzone-trust-untrust-outbound]policy 0
09:32:54  2014/04/03
[SRG-policy-interzone-trust-untrust-outbound-0]policy source 192.168.1.2 0
09:33:22  2014/04/03
[SRG-policy-interzone-trust-untrust-outbound-0]policy destination 10.1.1.2 0
09:33:37  2014/04/03
[SRG-policy-interzone-trust-untrust-outbound-0]policy service service-set ftp
09:33:43  2014/04/03
[SRG-policy-interzone-trust-untrust-outbound-0]action permit
09:33:46  2014/04/03
```

然后我们看看 FTP 客户端能否成功访问？





咦？怎么看不到服务器文件，看日志信息发现用户认证已经通过了，但是数据连接建立失败。

再检查一下配置吧：

Trust 和 Untrust 的域间缺省包过滤关闭，在 Outbound 方向配置了允许 PC 访问 FTP 服务器的一条安全策略。

```
[SRG]display policy interzone trust untrust inbound
10:16:52  2014/04/03
policy interzone trust untrust inbound
 firewall default packet-filter is deny
[SRG]display policy interzone trust untrust out
[SRG]display policy interzone trust untrust outbound
10:16:59  2014/04/03
policy interzone trust untrust outbound
 firewall default packet-filter is deny
 policy 0 (10 times matched)
  action permit
  policy service service-set ftp (predefined)
  policy source 192.168.1.2 0
  policy destination 10.1.1.2 0
```

查看会话表也成功建立了会话：

```
<SRG>display firewall session table
10:10:27  2014/04/03
 Current Total Sessions : 1
  ftp  VPN:public --> public 192.168.1.2:2068-->10.1.1.2:21
```

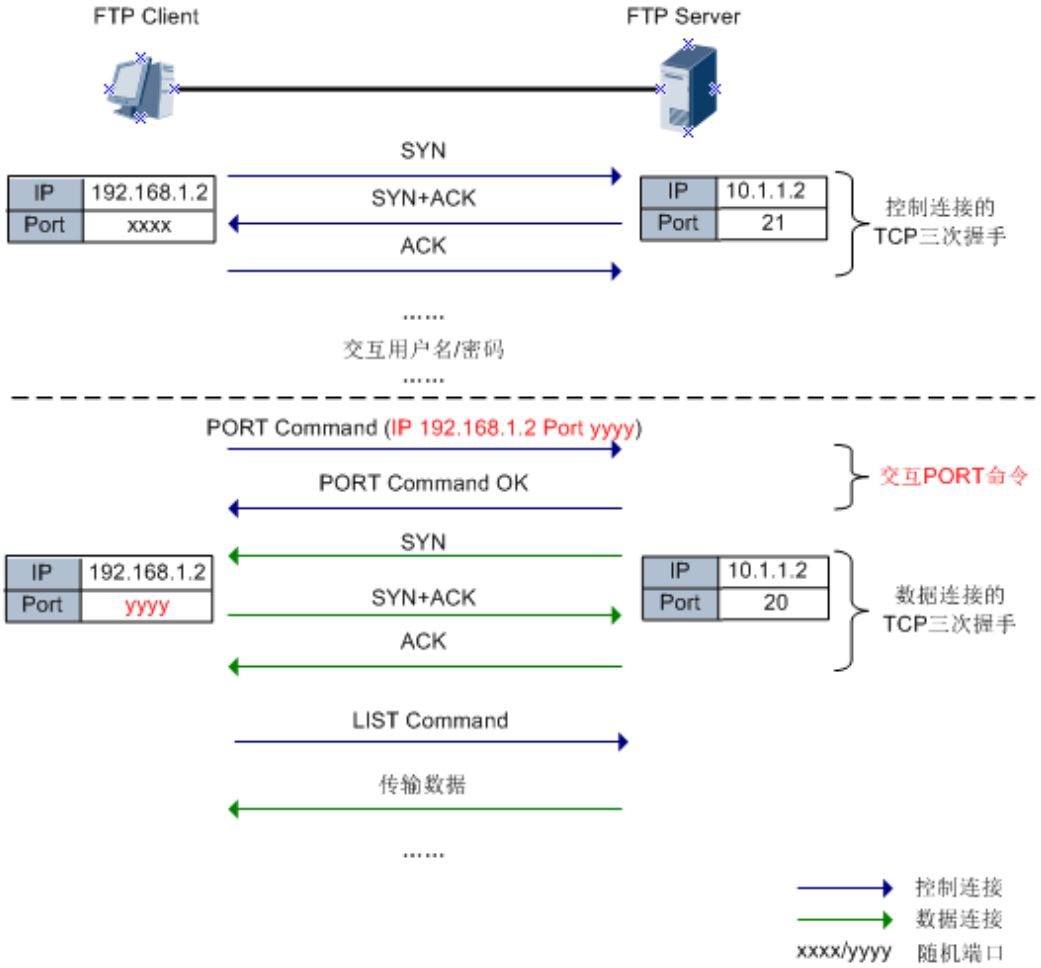
看起来应该没问题啊，不是说在首包的方向上应用安全策略，后续包直接匹配会话转发吗？

那我们分析一下 FTP 协议是否有什么特殊之处呢？

FTP 协议是一个典型的多通道协议，在其工作过程中，FTP Client 和 FTP Server 之间将会建立两条连接：控制连接和数据连接。控制连接用来传输 FTP 指令和参数，其中就包括建立数据连接所需要的信息；数据连接用来获取目录及传输数据。数据连接使用的端口号是在控制连接中临时协商的。

根据数据连接的发起方式 FTP 协议分为两种工作模式：主动模式（PORT 模式）和被动模式（PASV 模式）。主动模式中，FTP Server 主动向 FTP Client 发起数据连接；被动模式中，FTP Server 被动接收 FTP Client 发起的数据连接。

模式在一般的 FTP 客户端中都是可以设置的，这里我们以主动模式为例进行讲解，主动模式的协议交互流程如下：



首先 FTP 客户端向 FTP 服务器的 21 端口发起连接建立控制通道，然后通过 PORT 命令协商客户端使用的数据传输端口号。协商成功后，服务器主动向客户端的这个端口号发起数据连接。而且每次数据传输都会协商不同的端口号。

而我们配置的安全策略仅开放了 FTP 协议，也就是 21 端口。当 FTP 客户端向服务器发起控制连接时建立了如下会话。

```
<SRG>display firewall session table
10:10:27  2014/04/03
 Current Total Sessions : 1
  ftp  VPN:public --> public 192.168.1.2:2068-->10.1.1.2:21
```

而服务器向客户端发起数据连接的源/目的端口号分别是 20 和临时协商的端口号 yyyy，显然不是这条连接的后续报文，无法命中此会话转发。因此会出现可以验证用户密码，但是无法获取目录列表的现象。

有同学可能想到了，在服务器到客户端的方向也配置安全策略就行了吧？对，这是一种方法，但是这样必须开放客户端的所有端口有安全隐患。要是有一种方法可以自动记录数据连接就

好了！

别急，万能的防火墙都能实现。这就是本期要出场的神秘人物 ASPF（Application Specific Packet Filter）。ASPF 是针对应用层的包过滤，其原理是检测通过设备的报文的应用层协议信息，记录临时协商的数据连接，使得某些在安全策略中没有明确定义要放行的报文也能够得到正常转发。

记录临时协商的数据连接的表项称为 Server-map 表[1]，这相当于在防火墙上开通了“隐形通道”，使得像 FTP 这样的特殊应用的报文可以正常转发。当然这个通道不是随意开的，是防火墙分析了报文的应用层信息之后，提前预测到后面报文的行为方式，所以才打开了这样的一个通道。

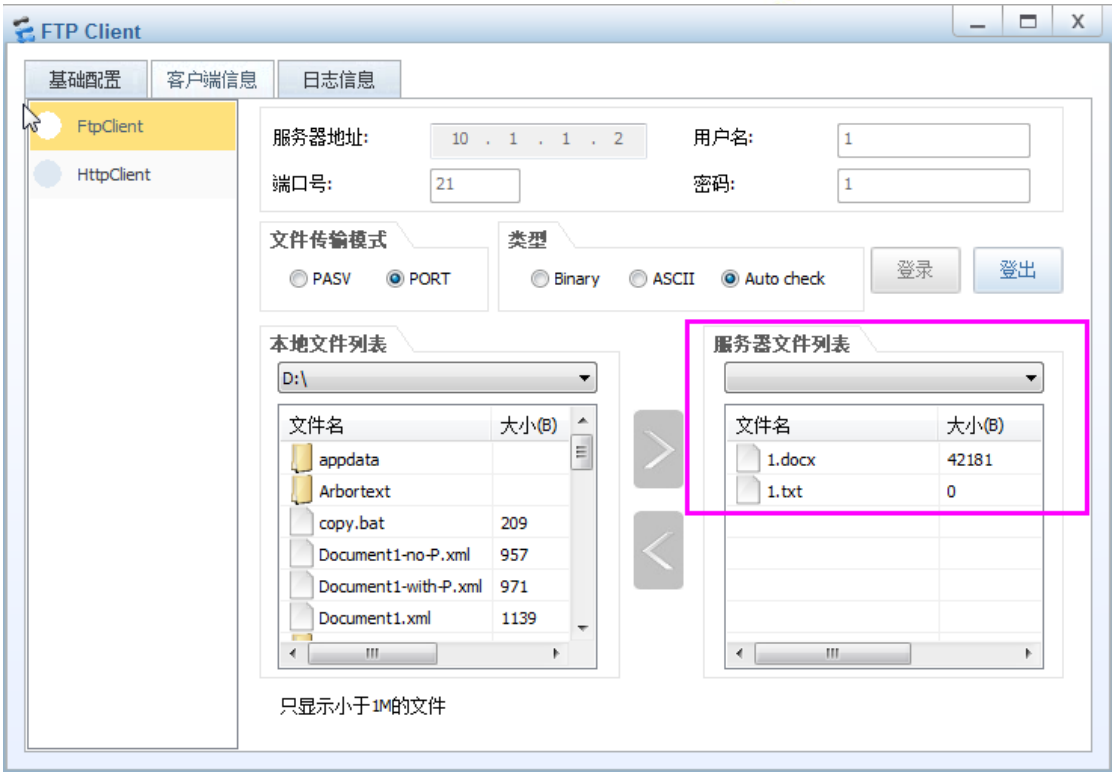
强叔解释

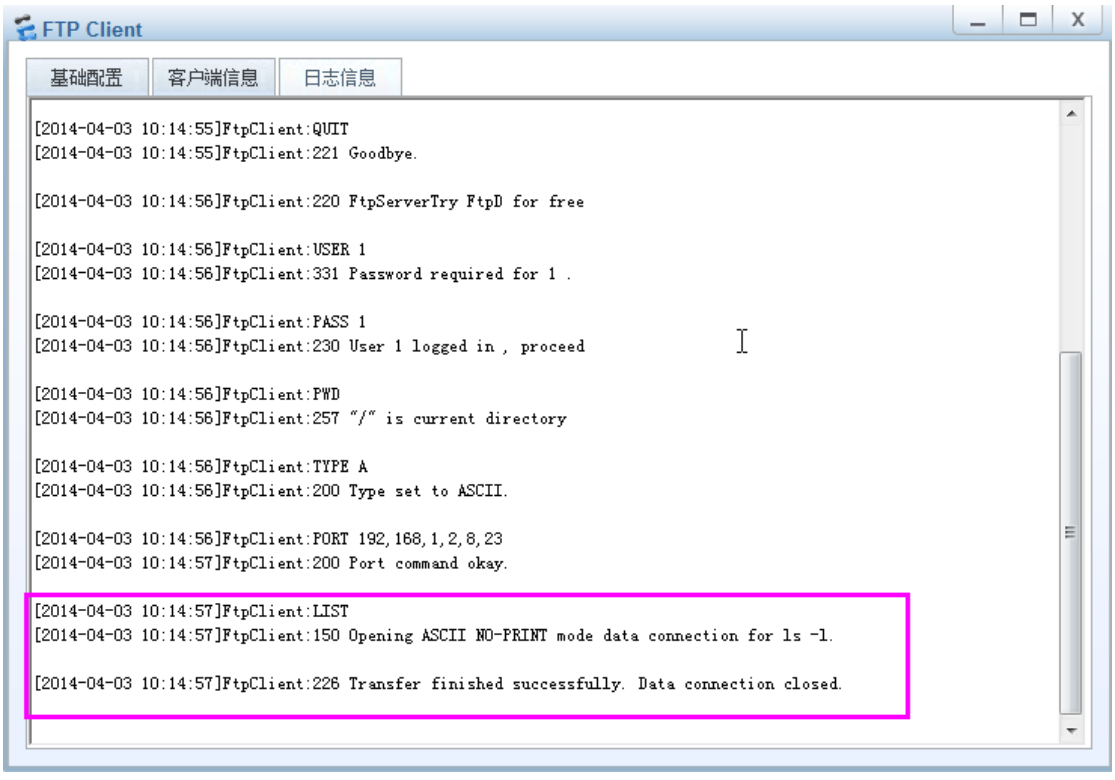
1：Server-map 表在防火墙转发中非常重要，不只是 ASPF 会生成，NAT Server、SLB 等特性也会生成 Server-map 表，后续在其他帖子中强叔还会提及。

说了这么多 ASPF 怎么配置呢，很简单，在域间配置一条命令即可 detece protocol。

```
[SRG]firewall interzone trust untrust
10:14:16  2014/04/03
[SRG-interzone-trust-untrust]detect ftp
10:14:20  2014/04/03
[SRG-interzone-trust-untrust]
```

FTP 访问成功：





此时查看 Server-map，可以看到已经自动生成了维护 FTP 数据连接的表项：

```
[SRG]display firewall server-map
10:15:42  2014/04/03
 server-map item(s)
 --------------------------------------------------------------------------
 ASPF, 10.1.1.2 -> 192.168.1.2:2071[any], Zone: ---
   Protocol: tcp(Appro: ftp-data), Left-Time: 00:00:07, Addr-Pool: ---
   VPN: public -> public
```

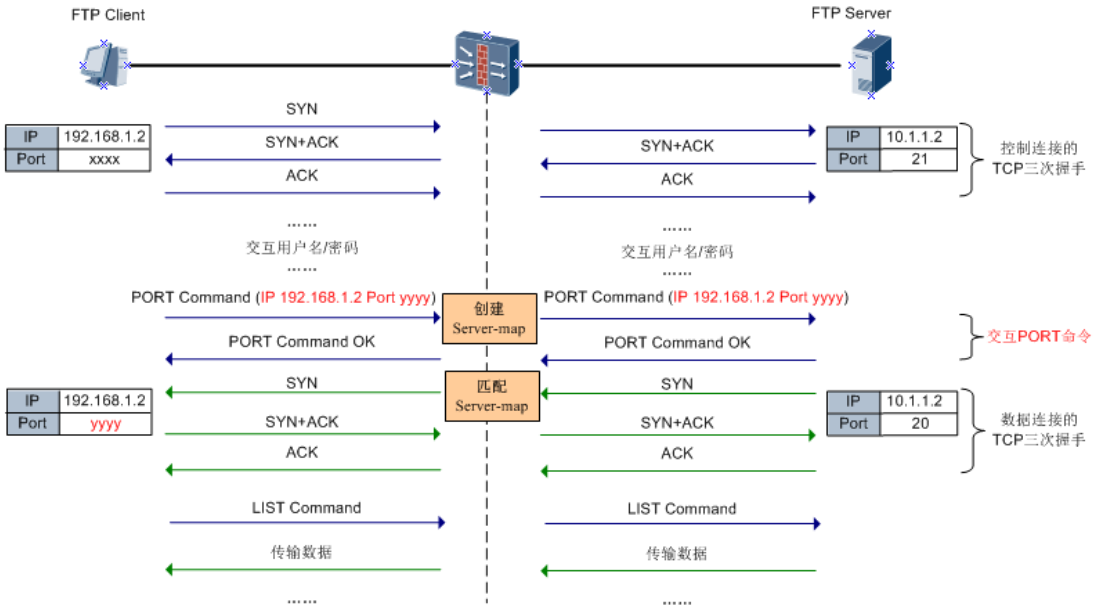
Server-map 表中记录了 FTP 服务器向 FTP 客户端的 2071 端口号发起的数据连接，服务器向客户端发起数据连接时将匹配这个 Server-map 表转发，而无需再配置反向安全策略。

数据连接的第一个报文匹配 Server-map 表转发后，防火墙将生成这条数据连接的会话，该数据连接的后续报文匹配会话表转发，不再需要重新匹配 Server-map 表项。

```
[SRG]display firewall session table
15:04:21  2014/04/04
 Current Total Sessions : 2
  ftp  VPN:public --> public 192.168.1.2:2053+->10.1.1.2:21
  ftp-data  VPN:public --> public 10.1.1.2:20-->192.168.1.2:2071
```

Server-map 表项由于一直没有报文匹配，经过一定老化时间后就会被删除。这种机制保证了 Server-map 表项这种较为宽松的通道能够及时被删除，保证了网络的安全性。当后续发起新的数据连接时会重新触发建立 Server-map 表项。

本期以 FTP 协议的主动模式为例做了讲解，FTP 的被动模式、其他多通道协议类似，不再一一讲解。总之就是配置了 ASPF 可以生成动态维护临时协商的数据连接的表项，既简化了安全策略的配置又确保了安全性。最后以一张图作为结束。

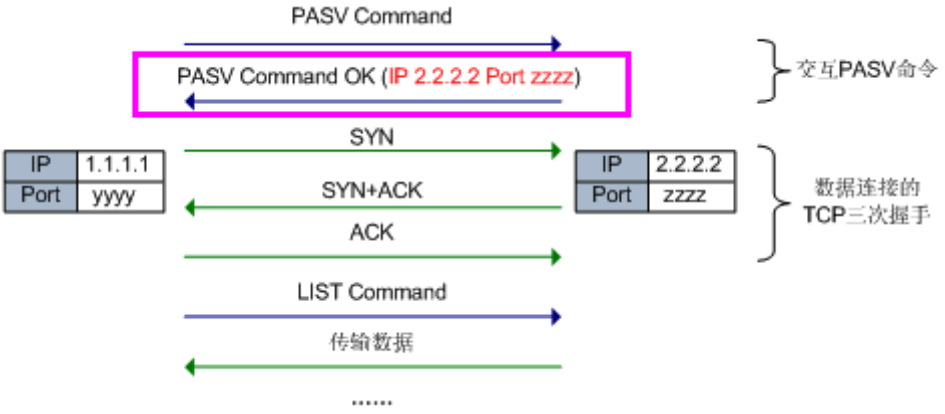


强叔提问

FTP 如果工作在被动模式（PASV）下，生成的 Server-map 表是啥样的呢？

已知：被动模式时动态协商的是服务器的端口号 zzzz，协商后客户端向服务器的这个端口号

发起数据连接。



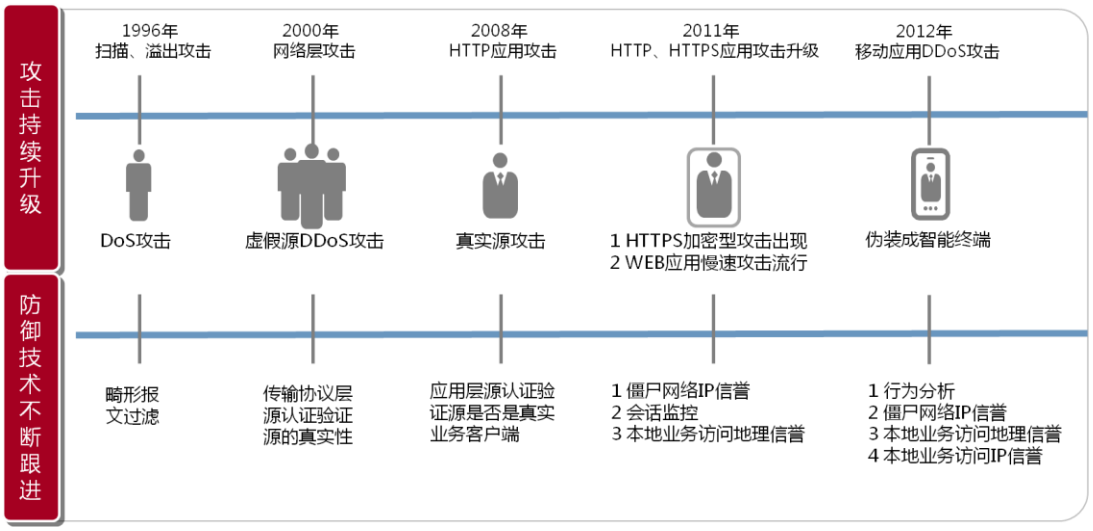
请在两个括号中填写内容：

ASPF: （） -> （） , Zone: ---
Protocol: tcp(Appro: ftp-data), Left-Time: 00:00:05, Addr-Pool: --

单包攻击及防御

大家好，强叔又和你见面了。前几期里，强叔带领大家了解了防火墙的基本安全特性，知道了防火墙的基本作用是保护特定网络免受“不信任”的网络的攻击，提到“网络攻击”，就不能不说说 DoS/DDoS 攻击，防范 DoS/DDoS 攻击是防火墙的基本安全功能。从今天开始，在接下来的几期里，强叔将带领大家一起去学习一下防火墙支持的单包攻击、流量型攻击和应用层攻击的防御。

九十年代，攻击随着互联网的蓬勃发展从实验室走向了 Internet。全球的攻击爱好者由于共同的信仰“Open Free Share（开源、免费、共享）”建立了同盟，很多年以后这帮人被叫做“黑客”。最初的黑客一般都是一些高级的技术人员，他们热衷于挑战、崇尚自由并主张信息的共享。随着 Internet 在全球的迅猛发展，政治、经济、军事、科技、教育、文化、生活等各个方面都逐渐网络化，信息已经成为物质和能量以外维持人类社会的第三资源，黑客也逐渐变成了一种有特殊目的的产业。



什么是 DoS 攻击？

DoS 是 Denial of Service 的简称，即拒绝服务，造成 DoS 的攻击行为被称为 DoS 攻击，其目的是使计算机或网络无法提供正常的服务。

那么，“拒绝服务”是什么意思呢？下面我们就打个形象的比喻。街边有一个小餐馆为大家提供餐饮服务，但是这条街上有一群地痞总是对餐馆搞破坏，比如：霸占着餐桌不让其他客人吃饭也不结账、或者堵住餐馆的大门不让客人进门，甚至骚扰餐馆的服务员或者厨师不让

他们正常干活，这样餐馆就没有办法正常营业了，这就是“拒绝服务”。Internet 中的计算机或者服务器就像是这个餐馆一样，为 Internet 用户提供互联网资源，如果黑客想要对这些计算机或者服务器进行 DoS 攻击的话，也使用消耗计算机或服务器性能、抢占链路带宽等手段！

最常见的 DoS 攻击就是我们常常提到的单包攻击。这类攻击一般都是以个人为单位的黑客发动的，攻击报文也比较单一，虽然破坏力强大，但是只要掌握了攻击的特征，防御起来还是比较容易的。

防火墙支持的单包攻击包括以下三大类：

- 畸形报文攻击：通常指攻击者发送大量有缺陷的报文，从而造成主机或服务器在处理这类报文时系统崩溃。
- 扫描类攻击：是一种潜在的攻击行为，并不具备直接的破坏行为，通常是攻击者发动真正攻击前的网络探测行为。
- 特殊控制报文攻击：也是一种潜在的攻击行为，不具备直接的破坏行为，攻击者通过发送特殊控制报文探测网络结构，为后续发动真正的攻击做准备。

单包攻击防御是防火墙具备的最基本的防范功能，华为全系列防火墙都支持对单包攻击的防御。下面强叔就带大家认识几种典型的单包攻击，以及华为防火墙是如何防范这些攻击的。

- Ping of Death 攻击及防御

操作系统处理数据包的大小是有限制的，IP 报文的长度字段为 16 位，即 IP 报文的最大长度为 65535。如果遇到大小超过 65535 的报文，会出现内存分配错误，从而使接收方的计算机系统崩溃。Ping of Death 攻击就是攻击者不断的通过 Ping 命令向攻击目标发送超过 65535

的报文，就可以使目标计算机的 TCP/IP 堆栈崩溃，致使接收方系统崩溃。

防火墙在处理 Ping of Death 攻击报文时，是通过判定数据包的大小是否大于 65535 字节，如果数据包大于 65535 字节，则判定为攻击报文，直接丢弃。

- Land 攻击及防御

Land 攻击是指攻击者向受害者发送伪造的 TCP 报文，此 TCP 报文的源地址和目的地址同为受害者的 IP 地址。这将导致受害者向它自己的地址发送回应报文，从而造成资源的消耗。

防火墙在处理 Land 攻击报文时，通过检查 TCP 报文的源地址和目的地址是否相同，或者 TCP 报文的源地址是否为环回地址，如果是则丢弃。

- IP 地址扫描攻击

IP 地址扫描攻击是攻击者运用 ICMP 报文（如 Ping 和 Tracert 命令）探测目标地址，或者使用 TCP/UDP 报文对一定地址发起连接，通过判断是否有应答报文，以确定哪些目标系统确实存活着并且连接在目标网络上。

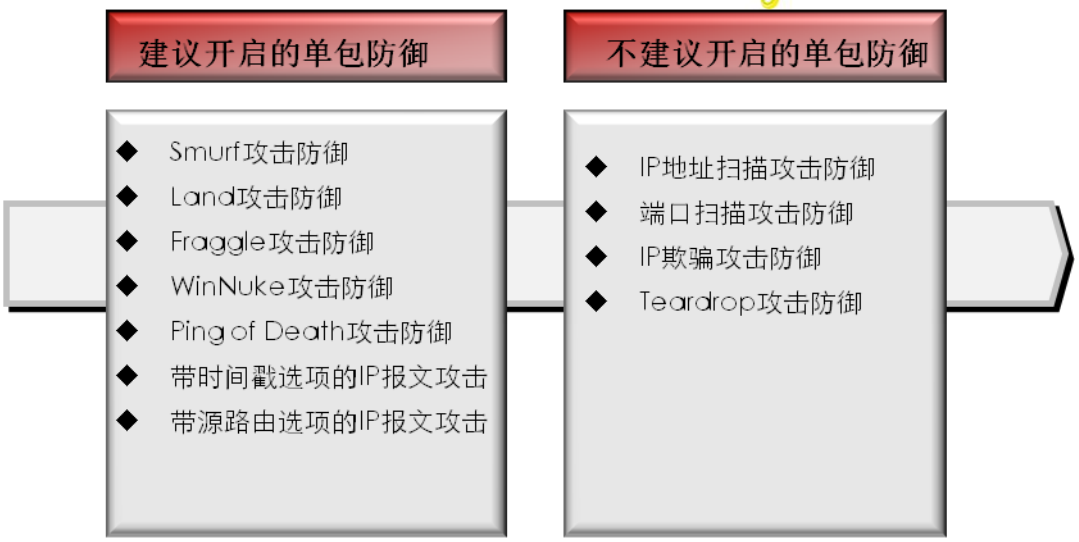
防火墙对收到的 TCP、UDP、ICMP 报文进行检测，当某源 IP 地址连续发送报文的目的 IP 地址与前一个报文的目的 IP 地址不同时，则记为一次异常，当异常次数超过预定义的阈值时，则认为该源 IP 的行为为 IP 地址扫描行为，防火墙会将该源 IP 地址加入黑名单。

可以看出，IP 地址扫描攻击并没有直接造成什么恶劣后果，它只是一种探测行为，通常是为了后续发动破坏性攻击做准备，尽管如此，这种行为我们防火墙也不会放过的。

从以上几种攻击及防御手段，我们可以发现，单包攻击一般都具有明显的特征，所以防火墙在防御单包攻击时，只要匹配了攻击特征，就可以很容易防御。

配置建议

防火墙支持的单包攻击防御种类繁多，在现网使用过程中，哪些需要配置，或者哪些不建议配置一直困扰着大家。下面强叔就为大家列举一下，哪几种攻击建议开启，哪几种攻击的防御不建议开启？



建议开启的单包攻击防御一般是现网比较常见的攻击，这种攻击开启以后，防火墙可以很好的进行防御，对性能等方面没有影响。而扫描类攻击在防御过程中比较消耗防火墙的性能，所以不建议开启。

其实，单包攻击在现网中所占的比例并不高，现网中最主流，也让人们最头疼的攻击其实是DDoS 攻击。DDoS 攻击种类比较多，华为防火墙支持的 DDoS 攻击包括 SYN Flood、UDP Flood、ICMP Flood 等流量型攻击，以及 HTTP Flood、HTTPS Flood、DNS Flood 等应用层攻击,下一期强叔就为大家介绍 DDoS 攻击中最常见的 SYN Flood 攻击及防御,敬请期待。

强叔提问

大家现在每天都会上网，网络攻击和我们的生活息息相关，那么大家都听说或遇到过什么样的攻击呢？高手们有没有用过什么攻击工具模拟过攻击报文？

流量型攻击之 SYN Flood 及防御

大家好，强叔和你们又见面了！上一期强叔带着大家一起了解了单包攻击的基本防御知识，知道了单包攻击的几大类型，以及防火墙支持防御的攻击种类。但是，在现网中单包攻击只占了很小一部分比例，更多的攻击还是集中在流量型攻击和应用层攻击。本期强叔将继续为大家讲解一下现网上常见的流量型攻击。

过去，攻击者所面临的主要问题是网络带宽，由于较小的网络规模和较慢的网络速度的限制，攻击者无法发出过多的请求。虽然类似“Ping of Death”的攻击类型只需要较少量的包就可以摧毁一个没有打过补丁的操作系统，但大多数的 DoS 攻击还是需要相当大的带宽，而以个人为单位的黑客们很难消耗高带宽的资源。为了克服这个缺点，DoS 攻击者开发了分布式的攻击。

木马成为黑客控制傀儡的工具，越来越多的计算机变成了肉鸡，被黑客所利用，并变成了他们的攻击工具。黑客们利用简单的工具集合许多的肉鸡来同时对同一个目标发动大量的攻击请求，这就是 DDoS(Distributed Denial of Service)攻击。随着互联网的蓬勃发展，越来越多的计算机不知不觉的被利用变成肉鸡，攻击逐渐变成一种产业。

提起 DDoS 攻击，大家首先想到的一定是 SYN Flood 攻击。今天强叔就给大家详细说说 SYN flood 的攻击和防御。

最初的 SYN Flood 攻击类似于协议栈攻击，在当年的攻击类型中属于技术含量很高的“高大上”。当年由于系统的限制以及硬件资源性能的低下，称霸 DDoS 攻击领域很久。它与别人的不同在于，你很难通过单个报文的特征或者简单的统计限流防御住它，因为它“太真实”、“太常用”。

SYN Flood 具有强大的变异能力，在攻击发展潮流中一直没有被湮没，这完全是他自身的优秀基因所决定的：

- 单个报文看起来很“真实”，没有畸形。
- 攻击成本低，很小的开销就可以发动庞大的攻击。

2014 年春节期间，某 IDC 的 OSS 系统分别于大年初二、初六、初七连续遭受三轮攻击，最长的一次攻击时间持续将近三个小时，攻击流量峰值接近 160Gbit/s！事后，通过对目标和攻

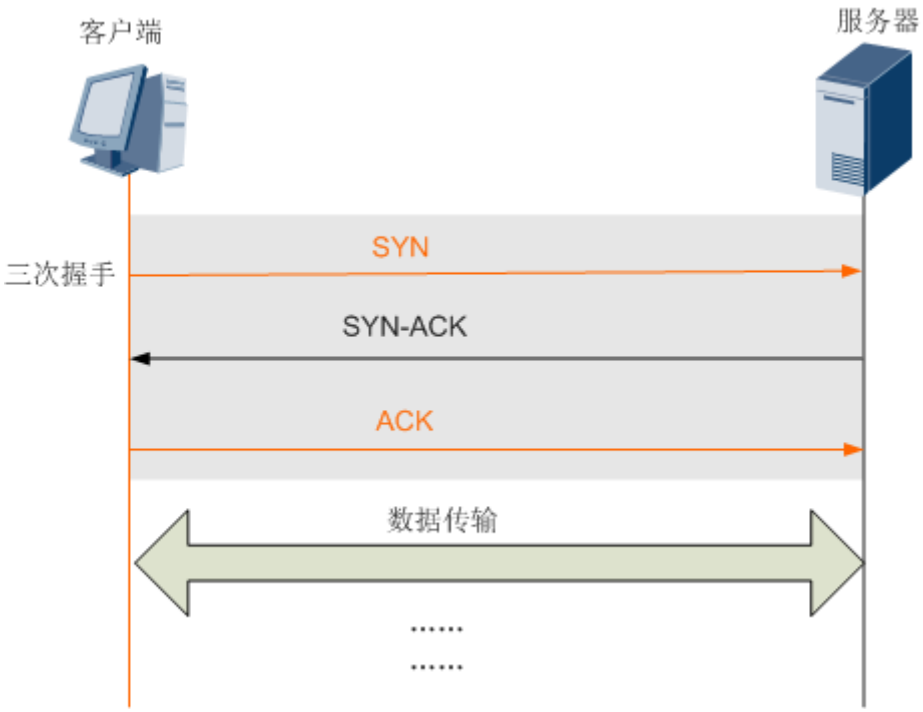
击类型分析，基本可以判断是有一个黑客/黑客组织发起针对同一目标的攻击时间。经过对捕获的攻击数据包分析，发现黑客攻击手段主要采用 SYN Flood。

2013 年，某安全运营报告显示，DDoS 攻击呈现逐年上升趋势，其中 SYN Flood 攻击的发生频率在 2013 全年攻击统计中占 31%。

可见，时至今日，SYN Flood 还是如此的猖獗。下面我们一起看一下它的攻击原理。

TCP 三次握手

SYN flood 是基于 TCP 协议栈发起的攻击，在了解 SYN flood 攻击和防御原理之前，还是要从 TCP 连接建立的过程开始说起。在 TCP/IP 协议中，TCP 协议提供可靠的连接服务，无论是哪一个方向另一方发送数据前，都必须先在双方之间建立一条连接通道，这就是传说中的 TCP 三次握手。



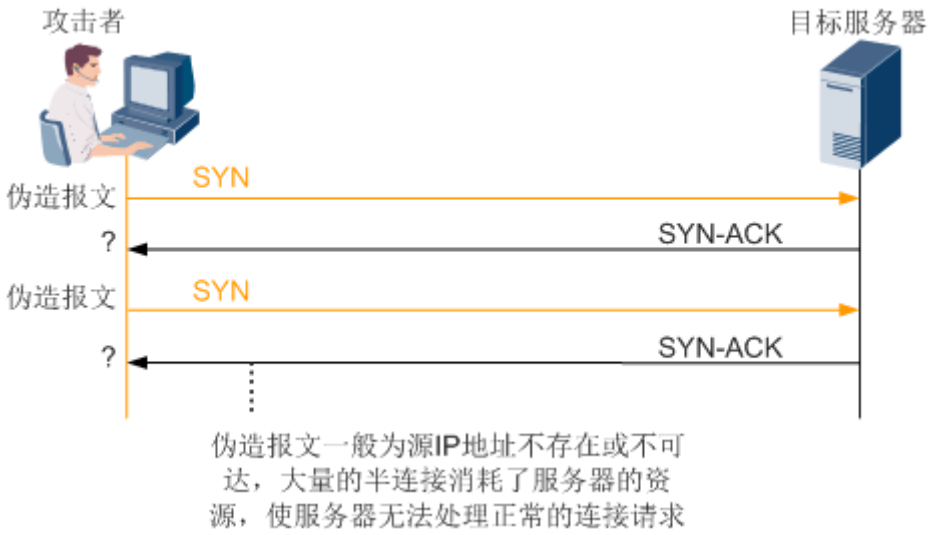
第一次握手：客户端向服务器端发送一个 SYN（Synchronize）报文，指明想要建立连接的服务器端口，以及序列号 ISN。

第二次握手：服务器在收到客户端的 SYN 报文后，将返回一个 SYN+ACK 的报文，表示客户端的请求被接受，同时在 SYN+ACK 报文中将确认号设置为客户端的 ISN 号加 1。ACK 即表示确认（Acknowledgment）。

第三次握手：客户端收到服务器的 SYN-ACK 包，向服务器发送 ACK 报文进行确认，ACK 报文发送完毕，三次握手建立成功。

如果客户端在发送了 SYN 报文后出现了故障，那么服务器在发出 SYN+ACK 应答报文后是无法收到客户端的 ACK 报文的，即第三次握手无法完成，这种情况下服务器端一般会重试，向客户端再次发送 SYN+ACK，并等待一段时间。如果一定时间内，还是得不到客户端的回应，则放弃这个未完成的连接。这也是 TCP 的重传机制。

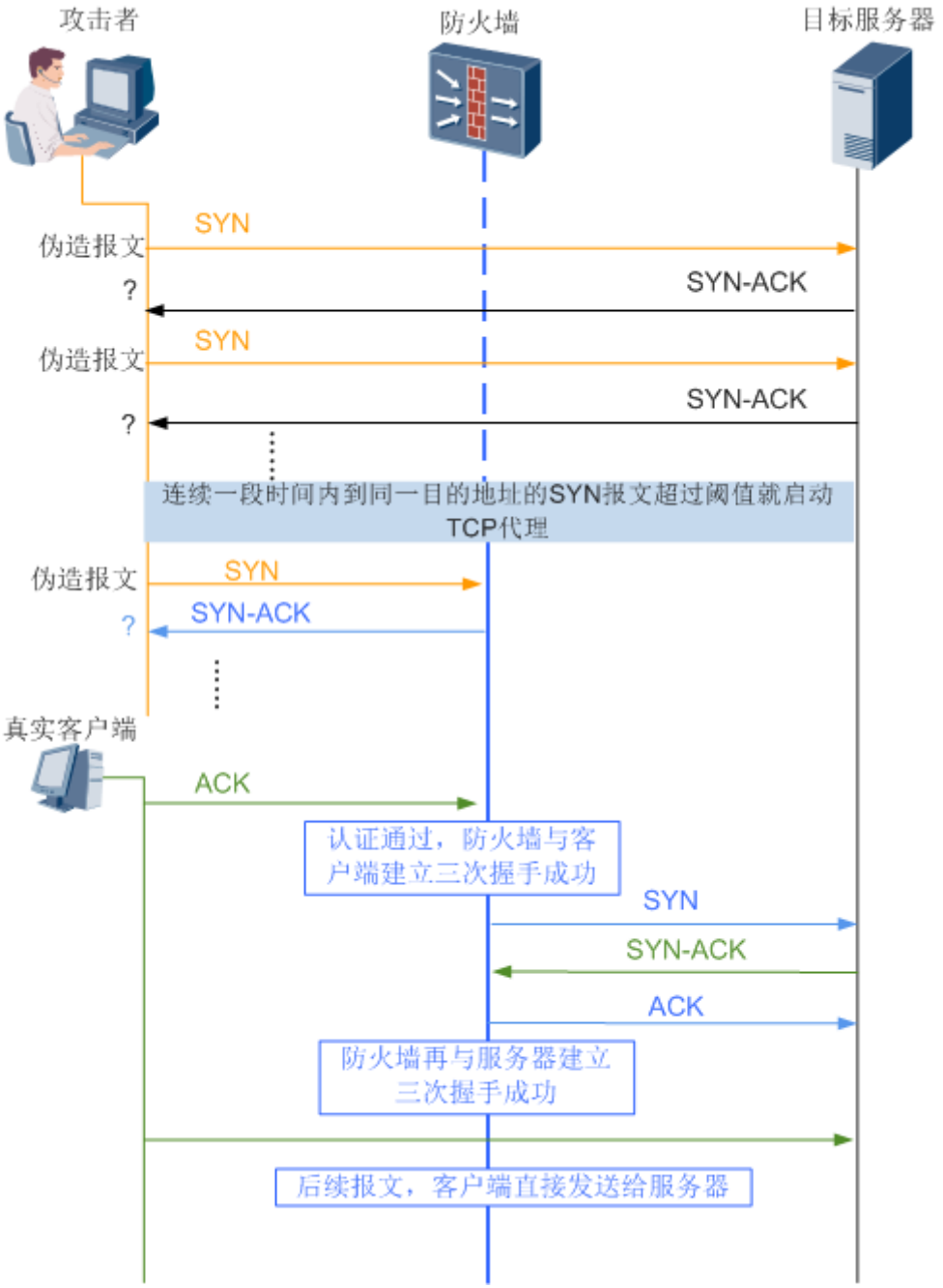
SYN Flood 攻击正是利用了 TCP 三次握手的这种机制。攻击者向服务器发送大量的 SYN 报文请求，当服务器回应 SYN+ACK 报文时，不再继续回应 ACK 报文，导致服务器上建立大量的半连接，直至老化。这样，服务器的资源会被这些半连接耗尽，导致正常的请求无法回应。



防火墙针对 SYN Flood 攻击，一般会采用 TCP 代理和源探测两种方式进行防御。

TCP 代理

TCP 代理是指我们的防火墙部署在客户端和服务器中间，当客户端向服务器发送的 SYN 报文经过防火墙时，防火墙代替服务器与客户端建立三次握手。一般用于报文来回路径一致的场景。



- 防火墙收到 SYN 报文，对 SYN 报文进行拦截，代替服务器回应 SYN+ACK 报文。
- 如果客户端不能正常回应 ACK 报文，则判定此 SYN 报文为非正常报文，防火墙代替服务器保持半连接一定时间后，放弃此连接。
- 如果客户端正常回应 ACK 报文，防火墙与客户端建立正常的三次握手，则判定此 SYN 报文为正常业务报文，非攻击报文。防火墙立即与服务器再建立三次握手，此连接的后续报文直接送到服务器。

整个 TCP 代理的过程对于客户端和服务器都是透明的。

TCP 代理过程中，防火墙会对收到的每一个 SYN 报文进行代理和回应，并保持半连接，所

以当 SYN 报文流量很大时，对防火墙的性能要求非常的高。

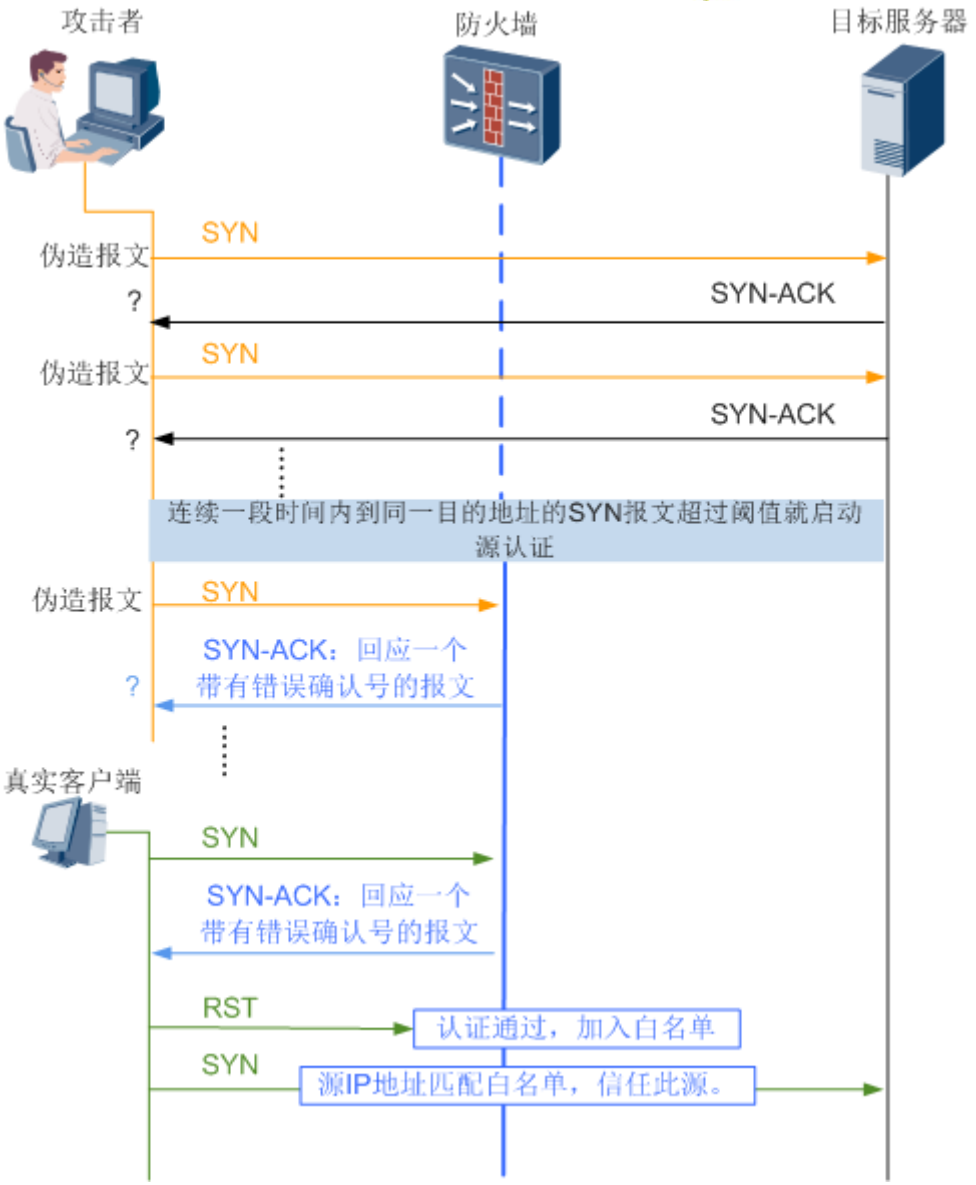
了解了攻击原理，再学习防御原理，是不是觉得很容易理解。讲了这么多，大家对 SYN Flood 的攻击以及 TCP 代理防御是不是有了一定的认识。其实，TCP 代理的本质就是利用防火墙的高性能，代替服务器承受半连接带来的资源消耗，由于防火墙的性能一般比服务器高很多，所以可以有效防御这种消耗资源的攻击。

但是 TCP 代理只能应用在报文来回路径一致的场景中，如果来回路径不一致，代理就会失败。可是在现网中，报文来回路径不一致的场景也是很常见的，那这种情况下如果发生了 SYN Flood 攻击，防火墙要怎么防呢？

不用担心，我们还有第二个杀手锏：TCP 源探测！

TCP 源探测

TCP 源探测是防火墙防御 SYN Flood 攻击的另一种方式，没有报文来回路径必须一致的限制，所以应用普遍。



- 当防火墙收到客户端发送的 SYN 报文时，对 SYN 报文进行拦截，并伪造一个带有错误序列号的的 SYN+ACK 报文回应给客户端。
- 如果客户端是虚假源，则不会对错误的 SYN+ACK 报文进行回应。
- 如果客户端是真实源发送的正常请求 SYN 报文，当收到错误的 SYN+ACK 报文时，会再发出一个 RST 报文，让防火墙重新发一个正确的 SYN+ACK 报文；防火墙收到这个 RST 报文后，判定客户端为真实源，则将这个源加入白名单，在白名单老化前，这个源发出的报文都认为是合法的报文，防火墙直接放行，不在做验证。

这里，我们再回头对比一下 TCP 源探测和 TCP 代理两种方式，会发现 TCP 源探测对客户端的源只做一次验证通过后，就加入白名单，后续就不会每次都对这个源的 SYN 报文做验证，这样大大提高了 TCP 源探测的防御效率和防御性能，可以有效缓解防火墙性能压力。

讲了这么多，大家是不是就会觉得 TCP 源探测对于 SYN Flood 已经是一个完美的防御方案了呢？它会不会也有什么弱点呢？

很长一段时间里，SYN Flood 在防火墙 TCP 代理和 TCP 源探测双重防御的压制下，得到了遏制。但是随着木马被广泛植入到更多的肉鸡，一个初级黑客简单操作就可以操纵动则上 G 流量的时候， SYN Flood 变得更加嗜血。TCP 代理和 TCP 源探测方式说到底都是使用防火墙牺牲自身的 CPU 不断的来解决问题。但是一旦海量低开销的 SYN Flood 攻击报文同时蜂拥而至时，这种伤敌一千自损八百的方式将走向另外一个极端，防火墙很有可能成为瓶颈。华为防火墙在不断提升自身性能的同时，还是可以承担一定的开销。但是传统的防御手段都是反弹，也就是说如果攻击流量是 1G 的话，防火墙的反弹流量也有 1G，这样就相当于有 2G 的“攻击”流量在互联网上占据着带宽，我们在防御的过程中不自觉的放大了垃圾流量，拥塞了链路。

魔高一尺，道高一丈，随着 SYN Flood 攻击的不断变异，防火墙也一直不断地提升着自身的防御能力。TCP 提供可靠的传输层，其中可靠性的保障之一就是确认从另一端收到的数据。但是数据和确认在传输过程中都有丢弃的可能，所以 TCP 通过在发送时设置一个定时器来解决这个问题。如果定时器到达设置的时间了，还是没有收到某个数据的确认报文，则 TCP 就会重传这个数据。华为专业 AntiDDoS 设备正是利用了 TCP 这种重传的机制，推出“首包丢弃”功能与“TCP 源探测”结合的防御方式，以应对超大流量的 SYN Flood 攻击。当 SYN 报文蜂拥而至时，专业 AntiDDoS 设备会将收到的第一个报文记录并直接丢弃，然后等待第二个重传报文。收到重传报文后，再对重传报文进行源探测。

这里提到了专业 AntiDDoS 设备，强叔也顺便给大家介绍一下，虽然防火墙具备 DDoS 防御能力，但是他毕竟不是专业的防攻击设备，术业有专攻，华为公司推出的 AntiDDoS1000 和 AntiDDoS8000 系列是专业的 AntiDDoS 设备，先进的防御技术在业界也是遥遥领先、大名鼎鼎，在腾讯、阿里巴巴都获得一致好评，是华为公司的尖刀产品哦！更多内容可以登录华为公司主页下载相关相产品文档。

强叔提问

回顾一下，在这一篇中，我们一起学习了 TCP 代理和 TCP 源探测，以及两种方式的利与避，还提到了防火墙只能在来回路径一致的场景中使用 TCP 代理，那么大家有没有想过，为什么防火墙在来回路径不一致的场景中，TCP 代理会失败呢？

流量型攻击之 UDP Flood 及防御

大家好，强叔又来了！上一期，强叔给大家介绍了 SYN Flood 的攻击和防御，本期强叔将带领大家一起来学习一下另一种常见的流量型攻击：UDP Flood。

讲 UDP Flood 之前，强叔还是先从 UDP 协议讲起。在讲 SYN Flood 的时候，我们知道了 TCP 协议是一种面向连接的传输协议。但是 UDP 协议与 TCP 协议不同， UDP 是一个无连接协议。使用 UDP 协议传输数据之前，客户端和服务器之间不建立连接，如果在从客户端到服务器端的传递过程中出现数据报的丢失，协议本身并不能做出任何检测或提示。因此，通常人们把 UDP 协议称为不可靠的传输协议。

既然 UDP 是一种不可靠的网络协议，那么还有什么使用价值或必要呢？

其实不然，在有些情况下 UDP 协议可能会变得非常有用。因为 UDP 具有 TCP 所望尘莫及的速度优势。虽然 TCP 协议中植入了各种安全保障功能，但是在实际执行的过程中会占用大量的系统开销，无疑使传输速度受到严重的影响。反观 UDP，由于排除了信息可靠传递机制，将安全和排序等功能移交给上层应用来完成，极大降低了执行时间，使传输速度得到了保证。

正是 UDP 协议的广泛应用，为黑客们发动 UDP Flood 攻击提供了平台。UDP Flood 属于带宽类攻击，黑客们通过僵尸网络向目标服务器发起大量的 UDP 报文，这种 UDP 报文通常为大包，且速率非常快，通常会造成以下危害：

- 消耗网络带宽资源，严重时造成链路拥塞。
- 大量变源变端口的 UDP Flood 会导致依靠会话转发的网络设备，性能降低甚至会话耗尽，从而导致网络瘫痪。

防火墙对 UDP Flood 的防御并不能像 SYN Flood 一样，进行源探测，因为它不建立连接。那应该怎么防御呢？

最初防火墙对 UDP Flood 的防御方式就是限流，通过限流将链路中的 UDP 报文控制在合理的带宽范围之内。

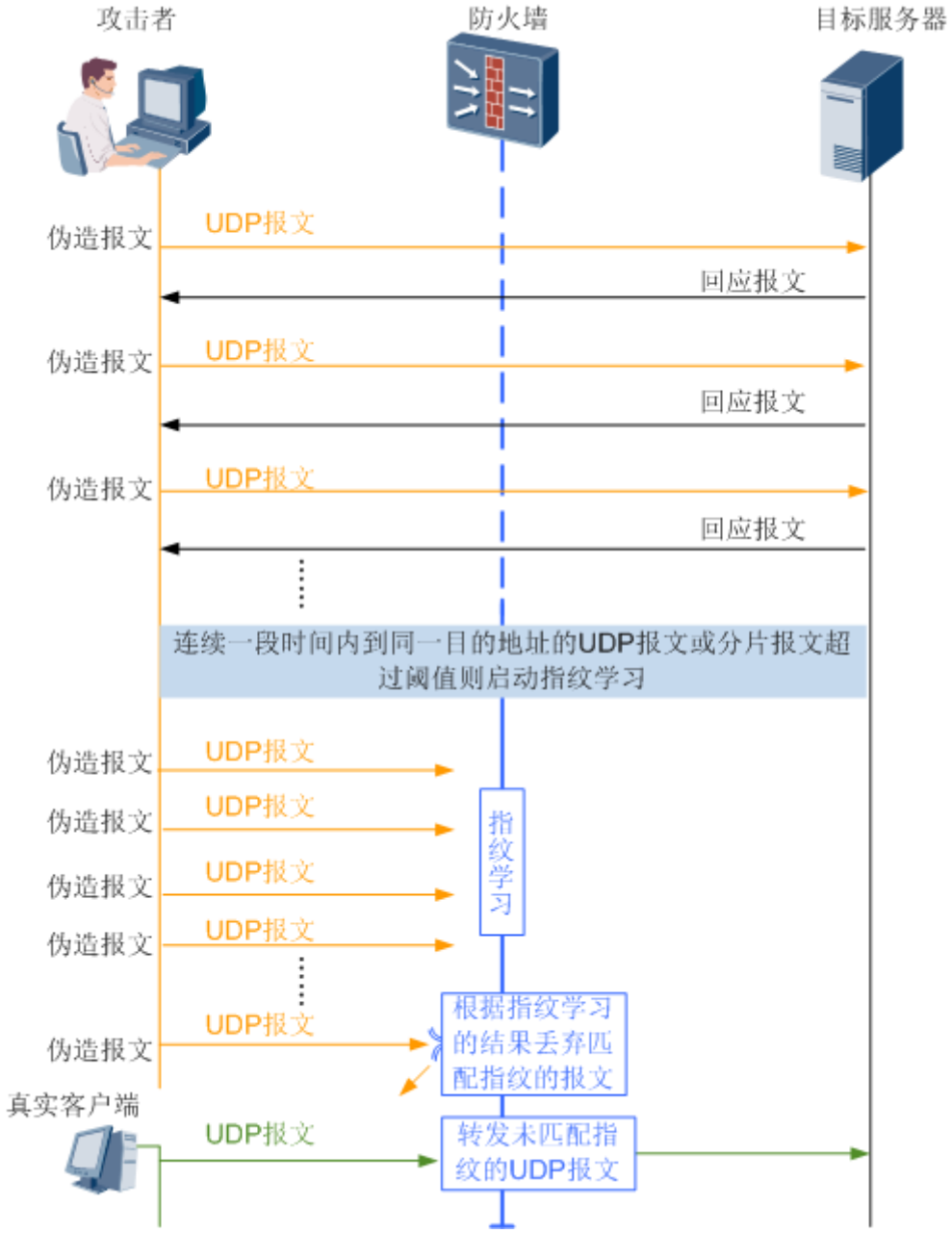
防火墙上针对 UDP Flood 的限流有三种：

- 基于目的 IP 地址的限流：即以某个 IP 地址作为统计对象，对到达这个 IP 地址的 UDP

流量进行统计并限流，超过部分丢弃。

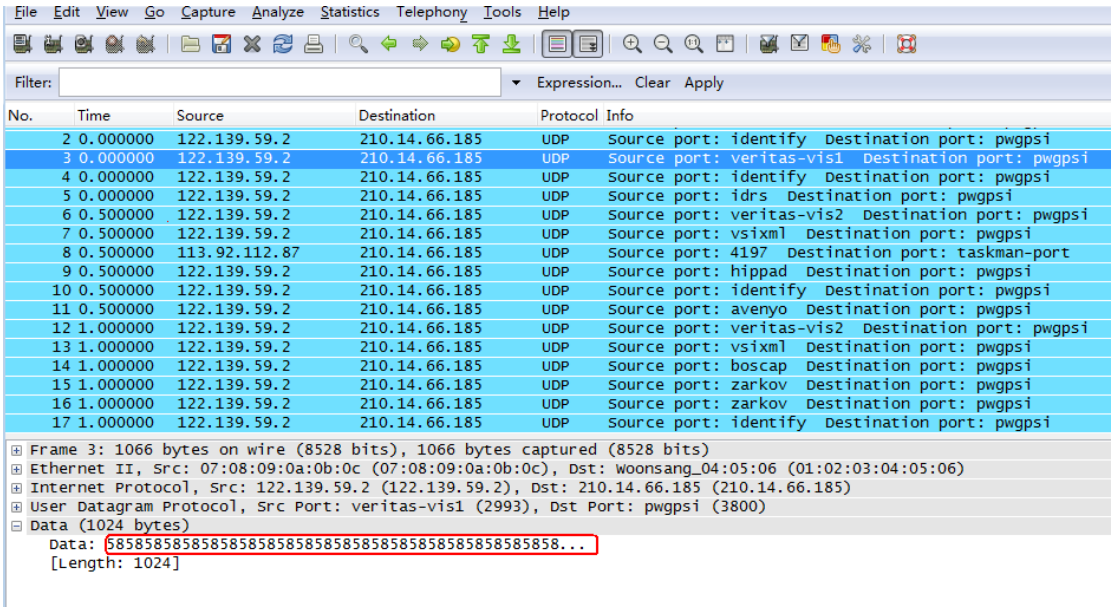
- 基于目的安全区域的限流:即以某个安全区域作为统计对象,对到达这个安全区域的 UDP 流量进行统计并限流，超过部分丢弃。
- 基于会话的限流：即对每条 UDP 会话上的报文速率进行统计，如果会话上的 UDP 报文速率达到了告警阈值，这条会话就会被锁定，后续命中这条会话的 UDP 报文都被丢弃。当这条会话连续 3 秒或者 3 秒以上没有流量时，防火墙会解锁此会话，后续命中此会话的报文可以继续通过。

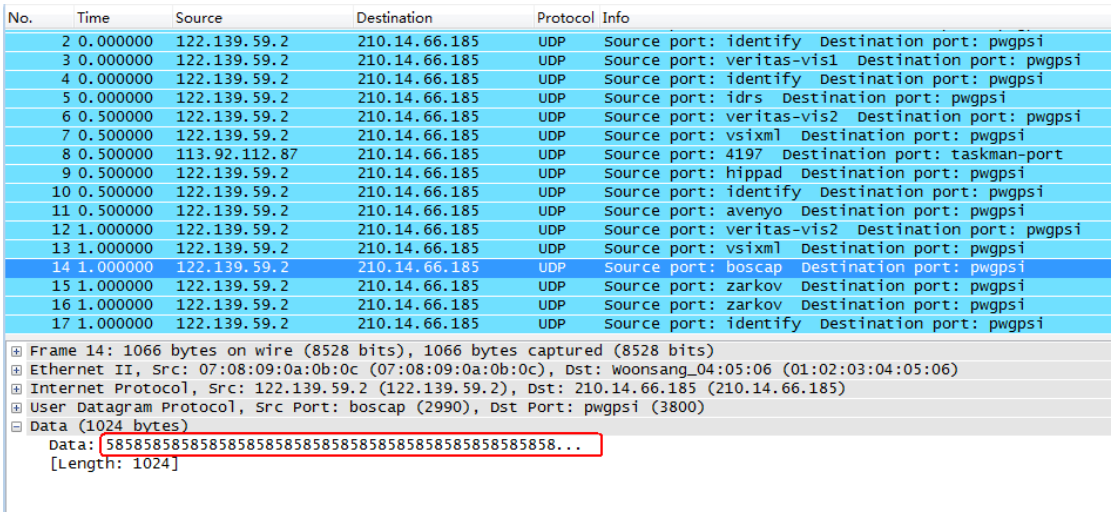
限流虽然可以有效缓解链路带宽的压力,但是这种方式简单粗暴,容易对正常业务造成误判。为了解决这个问题，防火墙又进一步推出了针对 UDP Flood 的指纹学习功能。



仔细分析，不难发现，UDP Flood 攻击报文具有一定的特点，这些攻击报文通常都拥有相同的特征字段，比如都包含某一个字符串，或整个报文内容一致。这些字段来自于 DDoS 工具自带的默认字符串，所以防火墙是通过收集这些字符串来检测 UDP Flood。这种防御算法在现网使用很多，主要因为黑客为了加大攻击频率，快速、长时间挤占攻击目标所在网络带宽，在使用攻击工具实现时直接在内存存放一段内容，然后高频发送到攻击目标，所以攻击报文具有很高的相似性。而正常业务的 UDP 报文一般每个报文负载内容都是不一样的，这样可以减少误判。

从下面的抓包中可以看出，到达相同目的 IP 地址的两个 UDP 报文的载荷是完全一样的，如果防火墙收到大量的类似这样的 UDP 报文，那么就有可能是发生了 UDP Flood 攻击。





指纹学习是通过分析客户端向服务器发送的 UDP 报文载荷是否有大量的一致内容，来判定这

个 UDP 报文是否异常。防火墙对到达指定目的地的 UDP 报文进行统计，当 UDP 报文达到告警阈值时，开始对 UDP 报文的指纹进行学习。如果相同的特征频繁出现，就会被学习成指纹，后续命中指纹的报文判定这是攻击报文，作为攻击特征进行过滤。

强叔再给大家总结一下，防火墙防御 UDP Flood 攻击主要有两种方式：限流和指纹学习，两种方式各有利弊。限流方式属于暴力型，可以很快将 UDP 流量限制在一个合理的范围内，但是不分青红皂白，超过就丢，可能会丢弃正常报文；而指纹学习属于理智型，不会随意丢弃报文，但是发生攻击后需要有个指纹学习的过程。目前，指纹学习功能是针对 UDP Flood 攻击的主流防御手段，在华为防火墙产品中广泛应用。

强叔提问

大家之前有没有配置过防火墙的 UDP Flood 防御功能？在配置过程中有遇到过什么样的问题吗？

应用层攻击及防御

随着计算机硬件的不断提升，运营商网络不断的扩容，我们的防火墙逐渐在线上增加，依靠抢占带宽制造的流量型攻击效果越来越差。黑客们开始寻求新的突破，转而向上进行应用层的攻击，应用层攻击逐渐变为黑客的焦点。应用层攻击比传输层攻击对于目标服务器造成更大的伤害。比如大型网站动态数据库链接的不断调用会比 SYN Flood 更加消耗系统的资源。今天强叔就给大家讲讲防火墙支持的 DNS Flood 和 HTTP Flood 攻击以及防御对策。

2009 年 5 月 19 日晚，江苏、安徽、广西、海南、甘肃、浙江等 6 省，分别报告省内域名递归解析服务因大量 DNS 请求陆续出现故障，其他多个省市则报告互联网域名解析服务出现异常，互联网运行受到严重影响，网络长时间处于断网状态。

让我们来回顾一下这次攻击事件的过程。5 月 19 日事发当晚，攻击者受利益驱使，对其他游戏“私服”网站的域名解析服务器 DNSPod 实施攻击，攻击流量超过 10G，导致 DNSPod 域名解析服务瘫痪。而 DNSPod 同时为暴风影音公司网站提供域名解析服务。

由于暴风影音软件中，有一项强制随机启动的名为 stormliv.exe 的进程，只要用户安装了暴风影音，该进程就会自动运行，并不断连接暴风影音网站，下载广告或升级。因此，当 DNSPod 服务器被攻击瘫痪时，数以千万计的暴风影音用户就充当了“肉鸡”，向运营商 DNS 递归服务器发送大量请求，DNS 访问流量瞬间就超过 30G，导致了本次重大网络安全事件。

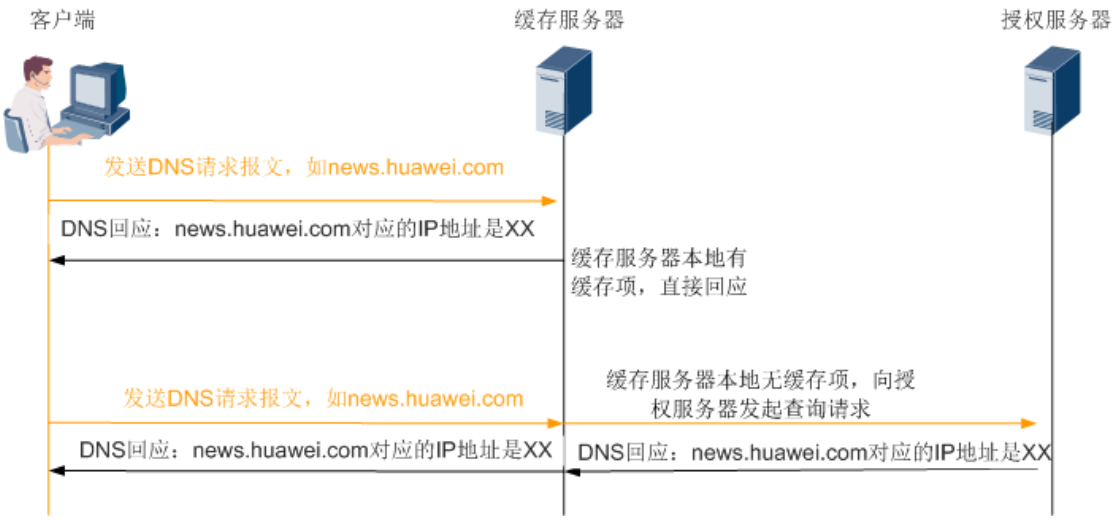
随后，公安机关介入侦查，攻击者于 5 月 29 日被抓获。调查发现，他们长期在互联网上经营游戏“私服”，并租用服务器专门协助他人攻击其他游戏“私服”和“私服”网站以谋取利益。

由此可见，应用层攻击造成的伤害是巨大的，直接会影响到我们的正常生活，加强应用层攻击防御刻不容缓。下面强叔就从 DNS Flood 讲起。

DNS Flood

通常情况下，我们在上网访问网页的时候，输入的网址都是域名，比如 www.huawei.com，这个请求的域名会发送到 DNS 缓存服务器，以请求其对应的 IP 地址。如果 DNS 缓存服务器上有此域名和IP地址的映射关系，DNS缓存服务器就会将查询到得IP地址返回给客户端。

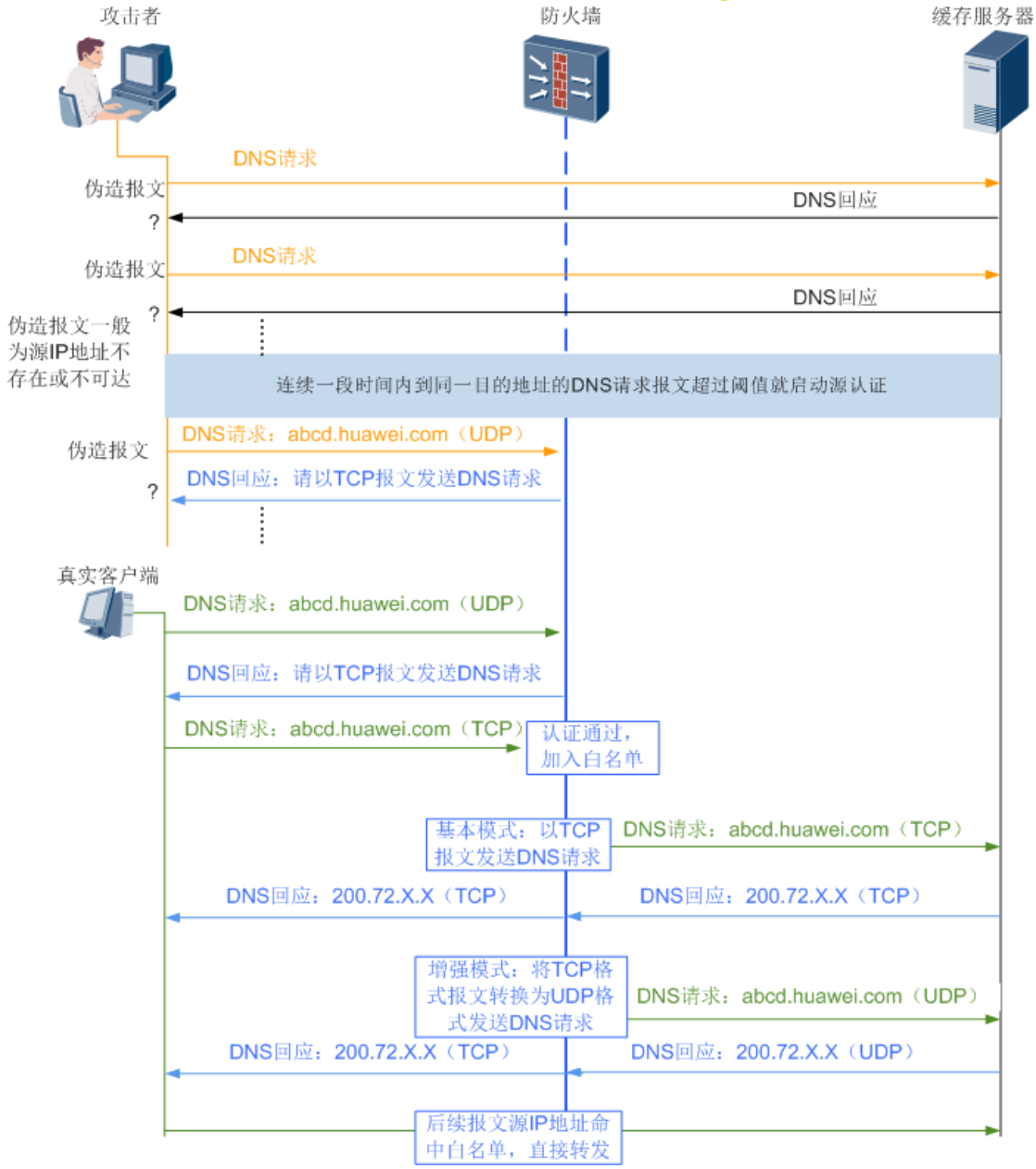
当 DNS 缓存服务器查找不到该域名与 IP 地址对应关系时，它会向授权 DNS 服务器发出域名查询请求。为了减少 Internet 上 DNS 的通信量，DNS 缓存服务器会将查询到的域名和 IP 地址对应关系存储在自己的本地缓存中。后续再有主机请求该域名时，DNS 缓存服务器会直接用缓存区中的记录信息回应，直到该记录老化，被删除。



常见的 DNS Flood 攻击一般都是攻击者向 DNS 服务器发送大量的不存在的域名解析请求，导致 DNS 缓存服务器不停向授权服务器发送解析请求，最终导致 DNS 缓存服务器瘫痪，影响对正常请求的回应。

我们先从 DNS 协议本身讲起。DNS 服务器支持 TCP 和 UDP 两种协议的查询方式，但是大多数的查询都是使用 UDP 查询的，我们都知道，UDP 提供无连接服务，传输速度快，可以降低服务器的负载。

也有特殊情况需要通过 TCP 方式查询，其中之一，就是 DNS 服务器可以设定使用 TCP 连接。当客户端向 DNS 服务器发起查询请求时，DNS 回应报文里有一个 TC 标志位，如果 TC 标志位置 1，就表示需要通过 TCP 方式查询。我们的防火墙就是利用这一机制对 DNS Flood 攻击进行防御。

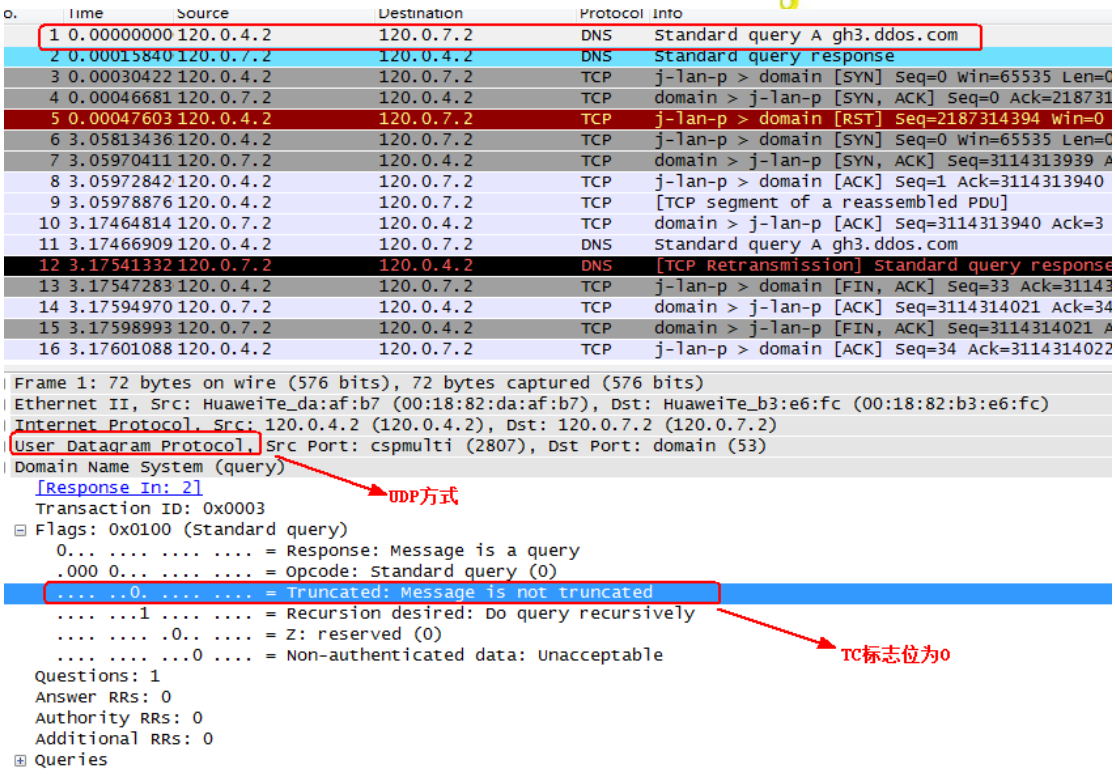


上图中，当发生 DNS Flood 攻击时，防火墙收到 DNS 请求，会代替 DNS 服务器响应 DNS 请求，并将 TC 标志位置 1，要求 DNS 客户端以 TCP 方式发送 DNS 请求。

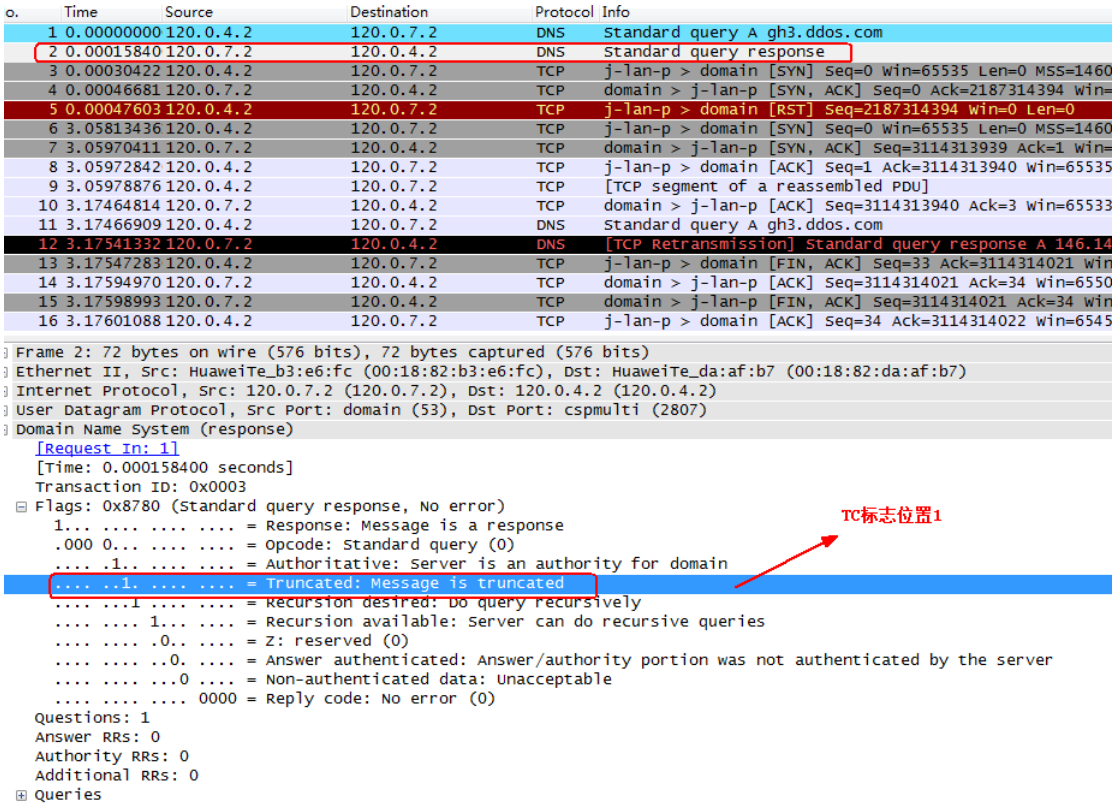
- 如果客户端是真实源，会继续以 TCP 方式发送 DNS 请求。
- 如果客户端是虚假源，则不会再以 TCP 方式发送 DNS 请求。

我们再来看一组真实客户端正常响应防火墙源探测的抓包信息：

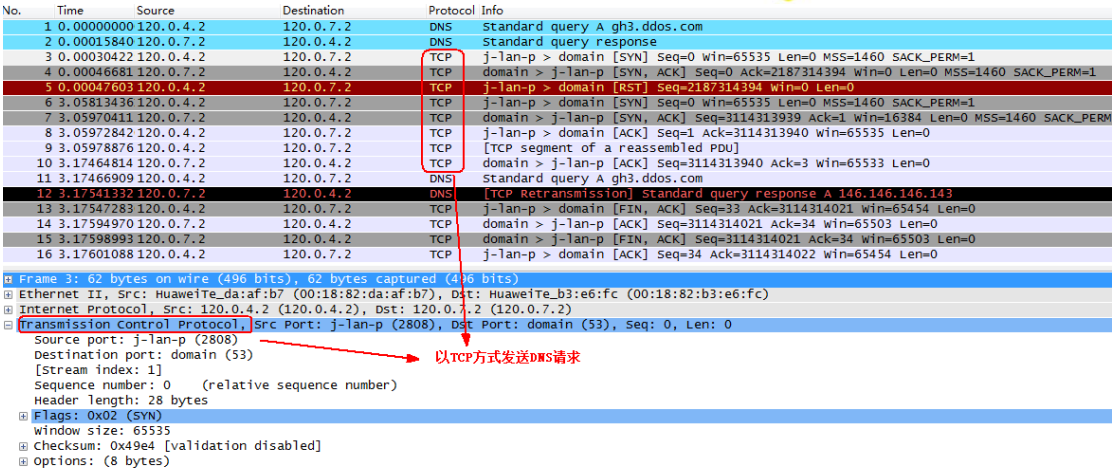
1. 客户端向 DNS 服务器以 UDP 方式发送查询请求。



2. 防火墙将 TC 标志位置 1，让客户端以 TCP 方式发送请求。



3. 客户端以 TCP 方式发送 DNS 请求。



这种方式可以很好的防御针对缓存服务器的 DNS 请求攻击，但是在现网使用过程中，并不是所有场景都适用。因为在源探测过程中，防火墙会要求客户端通过 TCP 方式发送 DNS 请求，但是并不是所有的客户端都支持以 TCP 方式发送 DNS 请求，所以这种方式在使用过程中也有限制。如果有正常客户端不支持以 TCP 方式发送 DNS 请求，使用此功能时，就会影响正常业务。

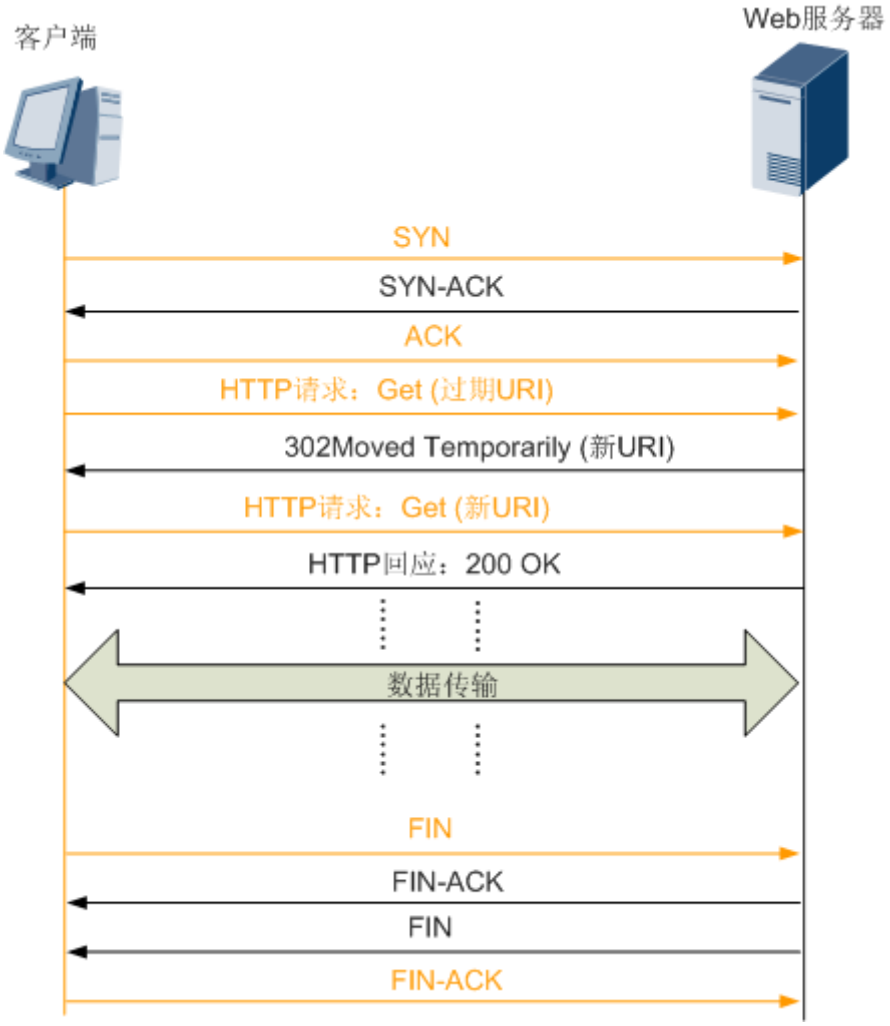
HTTP Flood

说完了 DNS Flood 攻击，强叔再给大家讲讲另一种常见的应用层攻击：HTTP Flood。近几年，HTTP Flood 攻击所占比例呈逐年上升趋势，不可小觑。

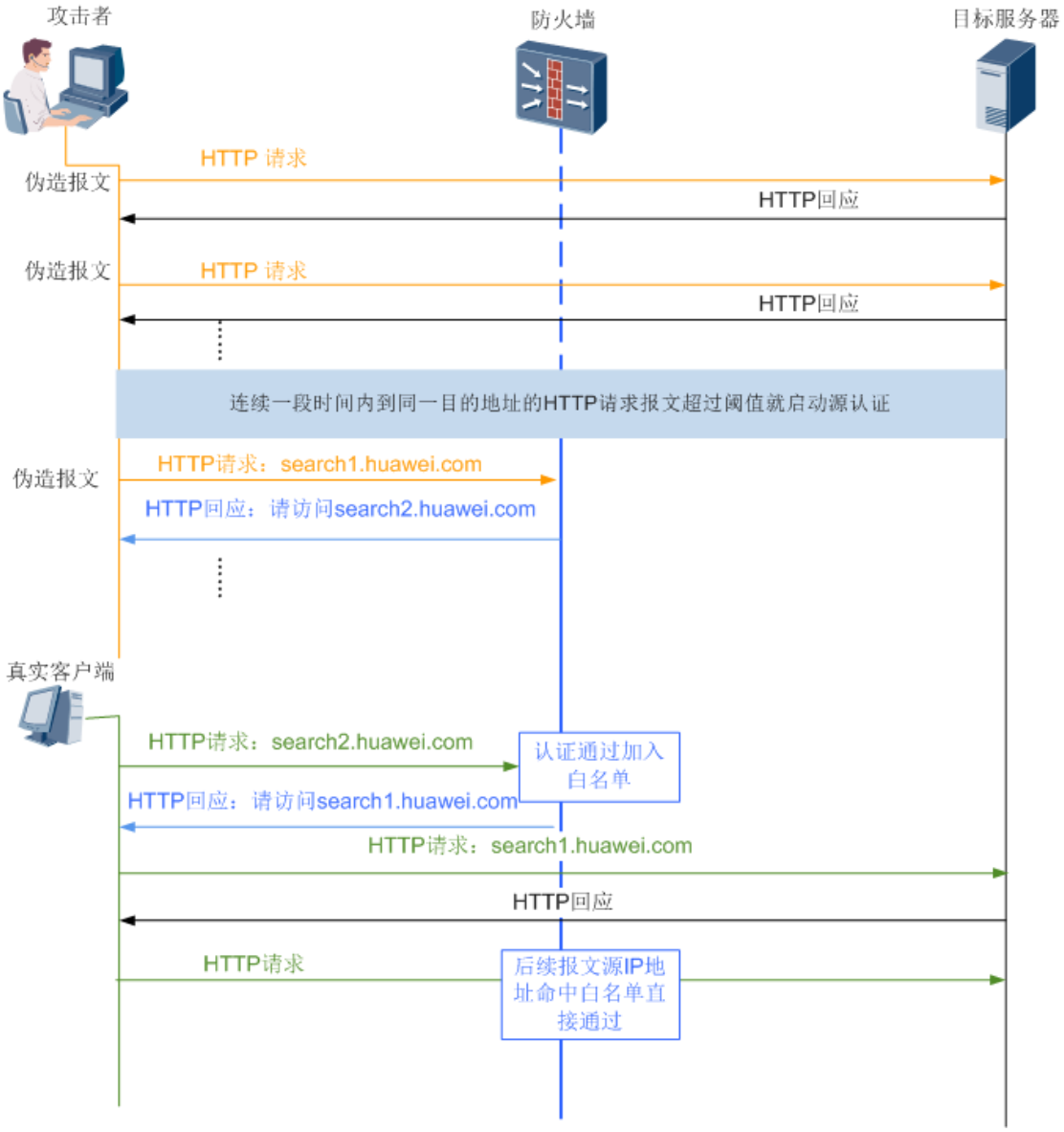
常见的 HTTP Flood 攻击，一般指黑客通过代理或僵尸主机向目标服务器发起大量的 HTTP 报文，这些请求涉及数据库操作的 URI 或其它消耗系统资源的 URI，目的是为了造成服务器资源耗尽，无法响应正常请求。

防火墙对于 HTTP Flood 的防御，主要依靠 HTTP 协议所支持的重定向方式，譬如说客户端向服务器请求 www.sina.com，服务器可以返回一个命令，让客户端改为访问 www.sohu.com。这种重定向的命令在 HTTP 协议栈中是合法的。我们防火墙的防御机制就是利用这个技术点，来探测 HTTP 客户端是否为真实存在的主机。

HTTP 报文的正常重定向过程是这样的：



防火墙正是利用了 HTTP 报文的这种重定向机制，在防御 HTTP Flood 攻击过程中，向客户端重定向一个其他的 URI。



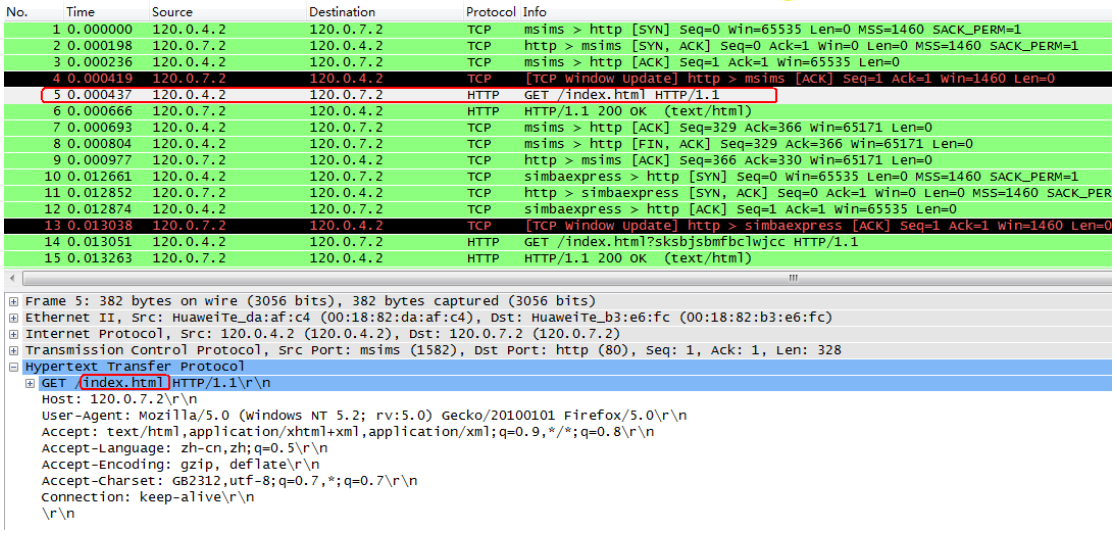
上图可以看出，当客户端访问 search1.huawei.com 的时候，防火墙重定向了一个 search2.huawei.com 地址给客户端让它访问：

- 如果客户端是虚假源，在收到防火墙发送的重定向地址后，不会重新发送 HTTP 请求。
- 如果客户端是真实源，则会对防火墙的重定向报文进行响应，并重新向 search2.huawei.com 地址发送请求。这样，防火墙收到 search2.huawei.com 请求后，即可判定这个客户端是真实源，并将这个客户端加入白名单。

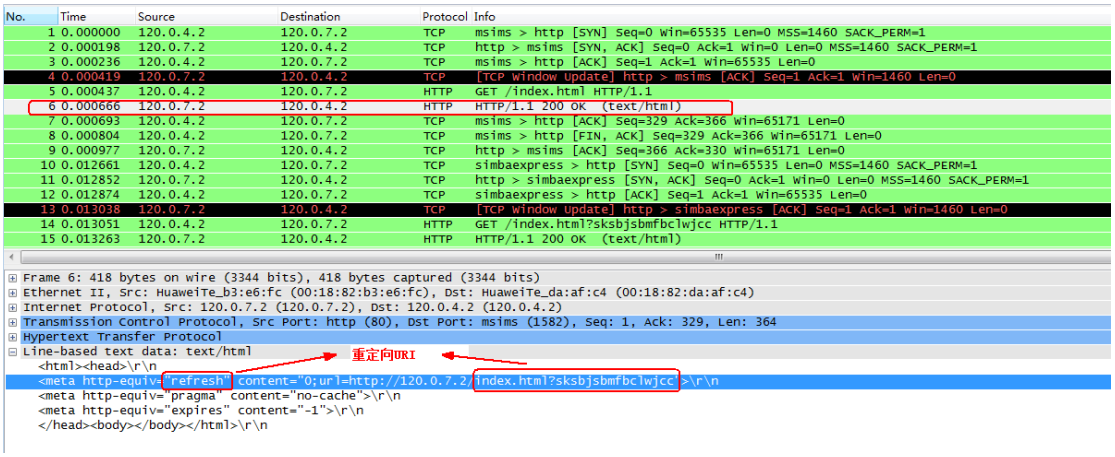
虽然在整个过程中，客户端需要自动重定向两次，但是时间是很短的，不会影响客户体验。

我们再看一组真实客户端正常响应防火墙重定向请求的抓包信息：

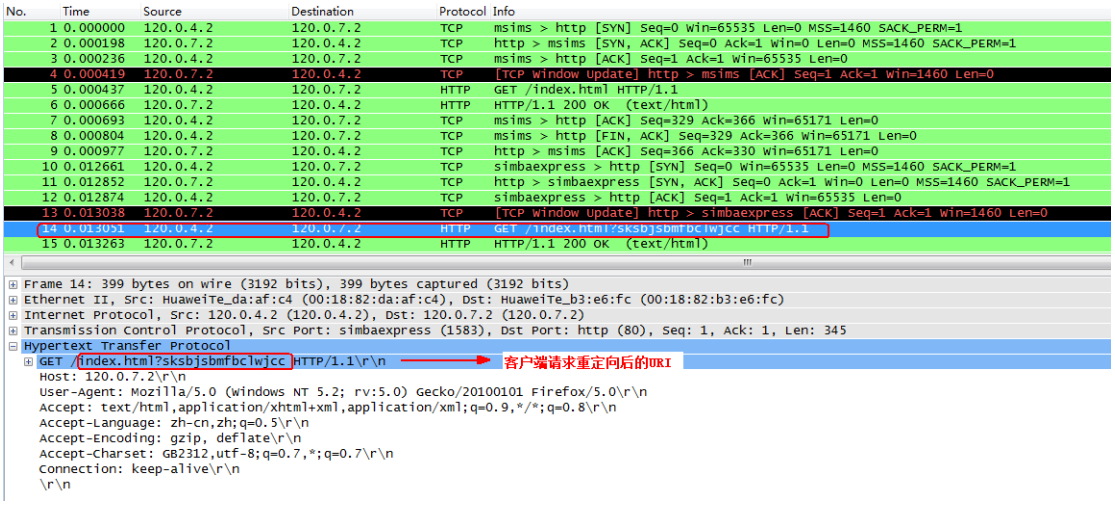
1. 客户端请求包含 URI 为“/index.html”的页面。



2. 防火墙对请求报文进行确认，并重定向一个新的 URI“/index.html?sksbjsbmfbclwjcc”给客户端。

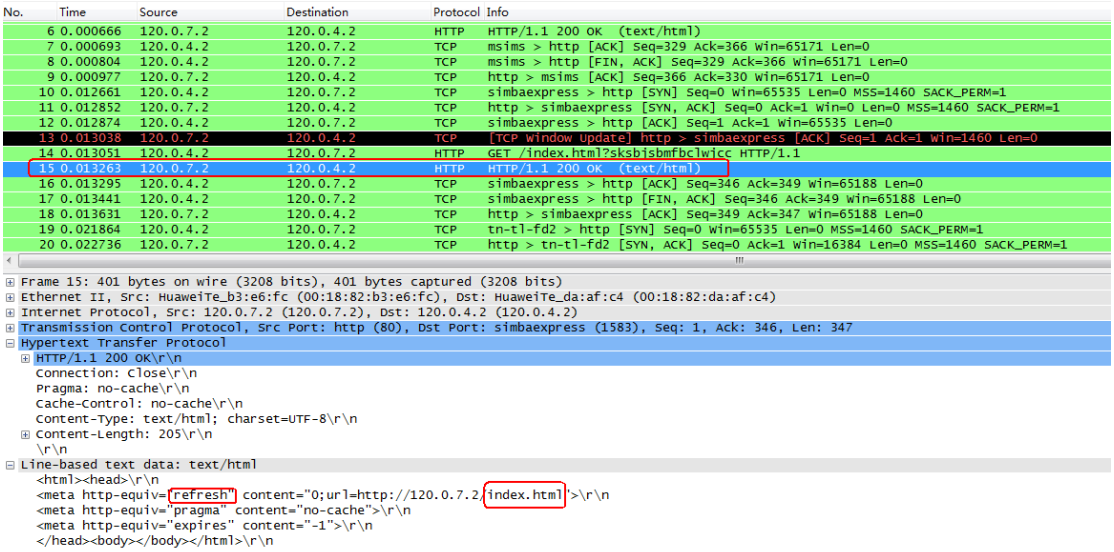


3. 客户端重新请求包含 URI 为“/index.html?sksbjsbmfbclwjcc”的页面。



4. 防火墙对包含新 URI 的请求进行确认，并将地址重新定向成包含 URI 为“/index.html”的页面。认证通过，后续客户端可以直接和服务器进行通信。整个过程对于用户来说是

透明的，重定向操作由客户端浏览器自动完成。



该模式可有效阻止来自非浏览器客户端的访问，如果僵尸工具没有实现完整的 HTTP 协议栈，不支持自动重定向，无法通过认证。而浏览器支持自动重定向，可以通过认证。这种防御方式在使用过程中，要确认客户端是否支持重定向。比如，常见的机顶盒就不支持自动重定向。所以在使用这种防御方式时，一定要确认所在的网络是否有机顶盒等客户端的正常业务，如果有，就不能使用此功能，否则会影响正常业务。

关于阈值怎么配置？

在这几期介绍的攻击防范中，大家对于 Flood 类攻击的阈值配置一直存在着疑惑，这里强叔给大家统一说明一下。

告警阈值配置要合理，如果配置过大，来攻击的时候就会防不住；如果配置过小，可能会把正常业务误判为攻击报文进行处理。

每个网络的流量模型都不同，配置阈值之前，需要有个前提准备，就是要大概了解这个网络正常情况下的每种类型报文的基本流量模型。这个值可能是管理员的一个经验值，也可以通过监测一段时间后通过学习得知。比如，我们想配置 SYN Flood 防御功能，配置告警阈值前，要先了解没有发生攻击的情况下网络中 SYN 报文的最大速率是多少，而 SYN Flood 防御的告警阈值一般可以配置为正常流量时的 1.2～2 倍。配置完告警阈值后，还要连续多观察几天，看这个阈值对正常业务是否有影响，如果有影响，要及时调整成更大的值。

现网中，攻击五花八门，黑客们也一直在不停的变换着攻击的花样，要想有效防御各种层出

不穷的攻击，防御手段也要不断的更新跟进。

术业有专攻，防火墙毕竟不是专业的 Anti-DDoS 设备，对现网复杂攻击的防御能力有限。如果想要更有效的防御 DDoS 攻击还要选择华为 Anti-DDoS 解决方案。华为 Anti-DDoS 解决方案对每一种攻击都具备精细化的防御功能。比如我们今天提到的 DNS Flood，

Anti-DDoS 解决方案是区分 DNS 缓存服务器和 DNS 授权服务器分别进行防御，除了可以防御 DNS Request Flood 和 DNS Replay Flood 攻击以外，还可以有效防御 DNS 投毒攻击、DNS 反射等多种类型的攻击。

华为 Anti-DDoS 解决方案提供业界领先的体系结构，包含检测中心、清洗中心和管理中心三个部分，具备很好的可扩展性，具有目前业界最高性能的 Anti-DDoS 设备。整个 Anti-DDoS 解决方案具备强大的攻击防御能力，七层防御技术层层过滤，可以专业防御 HTTP Flood、HTTPS Flood、DNS Flood、SIP Flood 等近百种的攻击，多样化的产品型号，满足高、中、低不同网络规模需求，在腾讯、阿里巴巴以及国内外知名运营商均获得一致好评！

想了解华为 Anti-DDoS 解决方案吗？拿起鼠标点击以下链接吧：

HUAWEI Secospace AntiDDoS8000 解决方案 产品文档

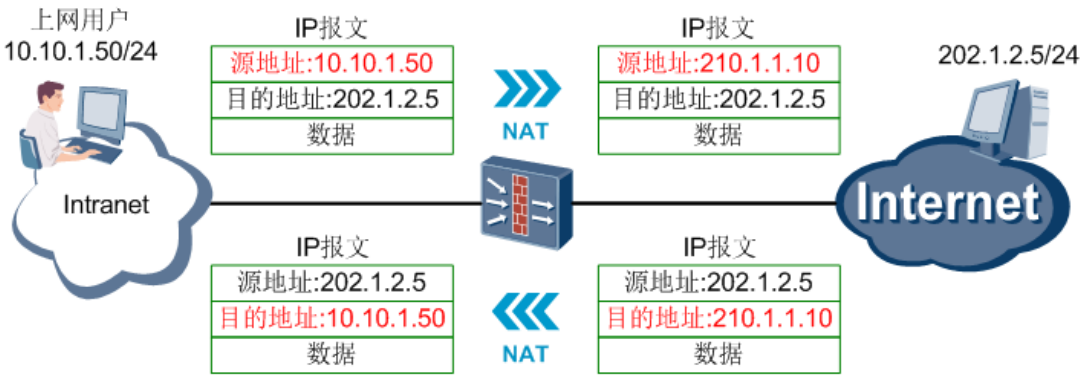
Anti-DDoS 攻击防御 原理描述

一墙当关，万夫上网——源 NAT（上篇）

当 Internet 技术初兴时，人们不会想到他会发展得如此迅猛，短短二十年已然深入到社会的方方面面。与此同时，很多之前没有考虑到的问题也都暴露出来，比如 IP 地址资源正在逐渐枯竭。人们在寻求 IPv4 替代方案的同时，也在积极研究各种技术来减少对 IP 地址的消耗，其中最出色的技术之一就是 NAT 技术（人们平时常说的 NAT 其实就是源 NAT）。

源 NAT 技术通过对报文的源地址进行转换，使大量私网用户可以利用少量公网 IP 上网，大大减少了对公网 IP 地址的需求。

下图示意了源 NAT 转换的过程：当上网流量到达防火墙时，报文的私网源 IP 将被转换为公网 IP；当回程报文到达防火墙时，报文的公网目的 IP 将被转换为私网 IP。整个 NAT 转换过程对于内、外网主机来说是完全透明的。



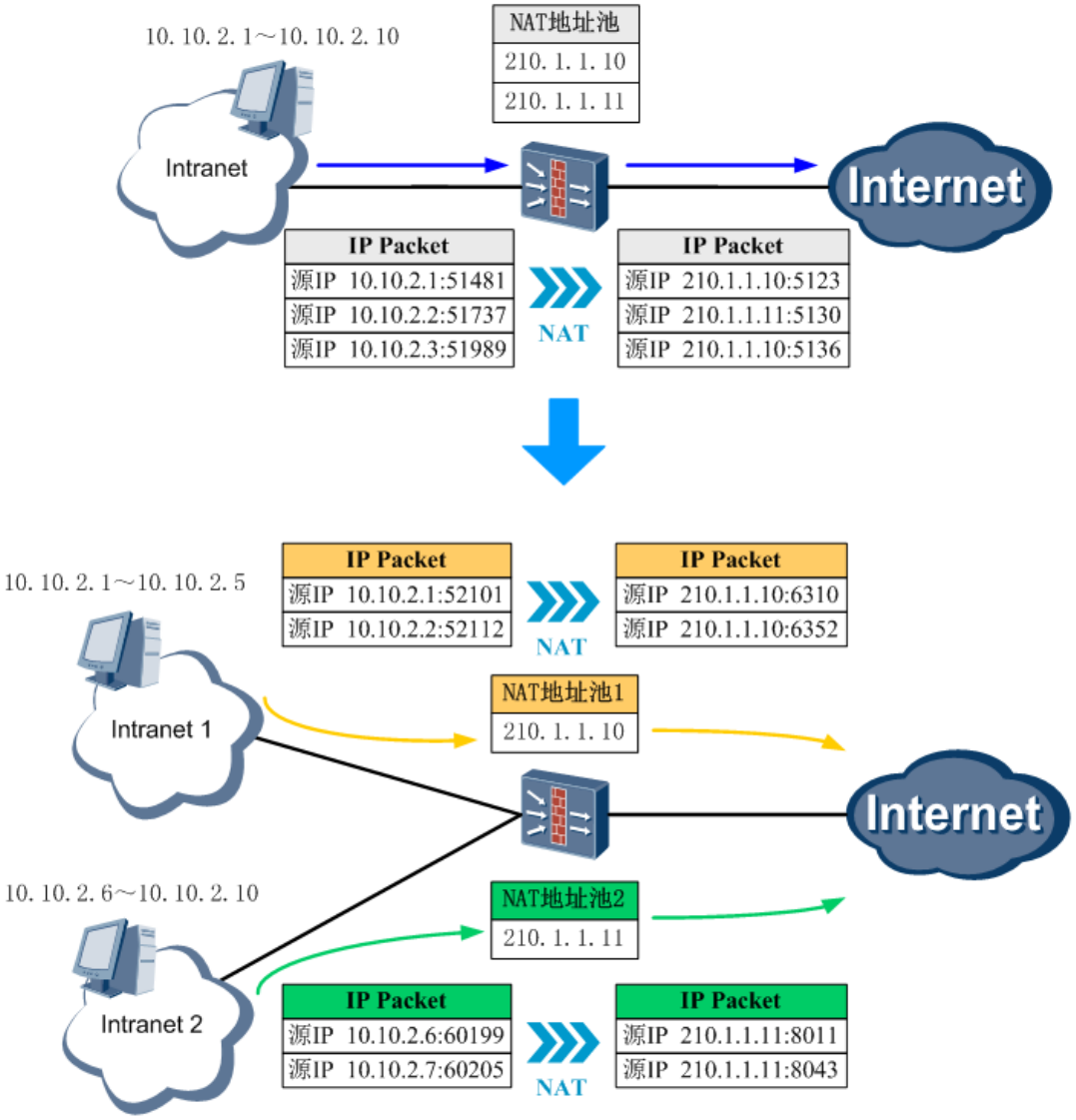
在介绍各种源 NAT 功能的特点和异同前，先介绍一下“NAT 地址池”。NAT 地址池是一个虚拟的概念，它形象地把“公网 IP 地址的集合”比喻成一个“放 IP 地址的池子或容器”，防火墙在应用源 NAT 功能时就是从地址池中挑选出一个公网 IP，然后对私网 IP 进行转换。**挑选哪个公网 IP 是随机的，和配置时的顺序、IP 大小等因素都没有关系。**

例 1 创建一个 NAT 地址池（以 eNSP 中的 USG5500 系列为例）

```
nat address-group 1    202.169.1.2    202.169.1.5
```

下面通过一个例子来说明地址池的使用方法。如下图所示，内网用户群（10.10.2.1-10.10.2.10）最初都在一个区域内，有两个公网 IP（210.1.1.10 和 210.1.1.11）可用于做 NAT 转换，由于无需对这些用户进行区分，所以可将 2 个公网 IP 放在同一地址池内。上网流量到达防火墙后，将从地址池中随机选取一个公网 IP 做 NAT 转换。

网络运行一段时间后，需要对用户进行区分，使用户群 1（10.10.2.1-10.10.2.5）和用户群 2（10.10.2.6-10.10.2.10）以不同的公网 IP 上网。由于 NAT 转换是随机选取公网 IP 的，所以 2 个公网 IP 在同一地址池内是无法满足此要求的。此时可将 2 个公网 IP 分别放在不同的地址池内，并指定用户群 1 使用地址池 1 做 NAT 转换，用户群 2 使用地址池 2 做 NAT 转换。这样，两个用户群做 NAT 转换后的 IP 就是不同的了。



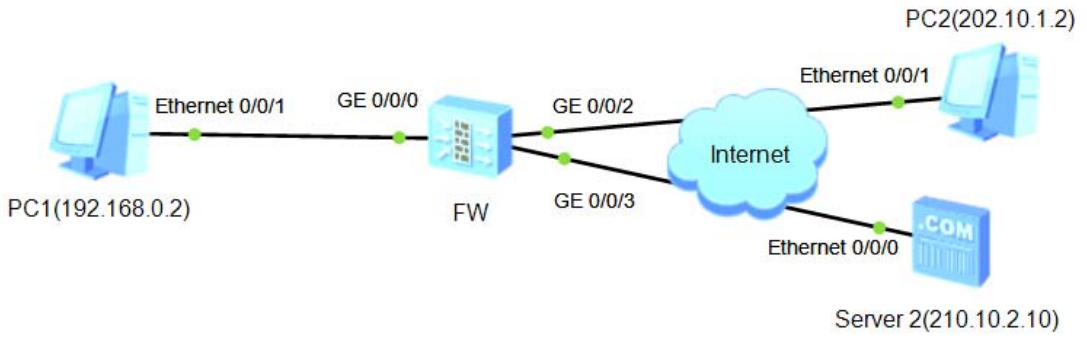
华为防火墙支持的源 NAT 功能如下表所示，且听强叔一一道来他们的特点和异同。

源 NAT 类型	私网 IP 和公网 IP 的数量对应关系	是否转换端口
NAT No-PAT	一对一	否
NAPT	多对一	是

源 NAT 类型	私网 IP 和公网 IP 的数量对应关系	是否转换端口
	多对多	
出接口地址方式（easy-ip）	多对一	是
Smart NAT （仅高端防火墙 USG9000 系列支持）	一对一 （预留 IP 做多对一转换）	否 （预留 IP 做端口转换）
三元组 NAT （仅高端防火墙 USG9000 系列支持）	多对一 多对多	是

NAT No-PAT

“No-PAT”表示不进行端口转换，所以 NAT No-PAT 只转换 IP 地址，故也称为“一对一 IP 地址转换”。我们使用如下组网进行演示：



在 FW 上配置源 NAT 模式，选择为 no-pat；将公网 IP 地址 202.30.1.1 和 202.30.1.2 加入 NAT 地址池 1；配置 NAT 策略，即对流量设置各种要求项，只有完全匹配上这些要求的流量才能利用 NAT 地址池 1 中的 IP 做 NAT 转换（**如果要针对源 IP 设置 NAT 策略，那么应该是做源 NAT 转换前的 IP**）。

例 2 配置 NAT No-PAT

```
#
 nat address-group 1 202.30.1.1 202.30.1.2
#
nat-policy interzone trust untrust outbound
 policy 1
```

```
  action source-nat
  policy source 192.168.0.0    0.0.0.255
  address-group 1 no-pat       //使用地址池 1 做 NAT No-PAT 转换
```

这里强叔要强调两个配置：安全策略和黑洞路由。

安全策略和 NAT 策略在字面上长的挺像，但是二者各司其职：安全策略检验是否允许流量通过，NAT 策略检验是否对流量进行 NAT 转换。由于防火墙检验流量是否符合安全策略的操作发生在检查 NAT 策略之前，所以**如果要针对源 IP 设置安全策略，则该 IP 应该是做源 NAT 转换前的 IP。**

例 3 配置安全策略

```
#
policy interzone trust untrust outbound
 policy 1
  action permit
  policy source 192.168.0.0    0.0.0.255
```

黑洞路由是一个让路由“有来无回”的路由，它的效果就是让设备丢弃命中该路由的报文。针对地址池中的公网 IP 必须配置黑洞路由，目的是防止产生路由环路。

例 4 配置黑洞路由

```
#
ip route-static 202.30.1.1    255.255.255.255 NULL0
ip route-static 202.30.1.2    255.255.255.255 NULL0
```

从 PC1 上 ping PC2，在 FW 上查看会话表和 Server-map 表。

从会话表中可以看到 PC1（192.168.0.2）的 IP 进行了 NAT 转换（**中括号[]内的是 NAT 转换后的 IP 和端口**），而端口没有转换。

```
[SRG]dis firewall session table

 Current Total Sessions : 5
  icmp  VPN:public --> public 192.168.0.2:54694[202.30.1.1:54694]-->202.10.1.2:20
48
  icmp  VPN:public --> public 192.168.0.2:54950[202.30.1.1:54950]-->202.10.1.2:20
48
  icmp  VPN:public --> public 192.168.0.2:55206[202.30.1.1:55206]-->202.10.1.2:20
48
  icmp  VPN:public --> public 192.168.0.2:55462[202.30.1.1:55462]-->202.10.1.2:20
48
  icmp  VPN:public --> public 192.168.0.2:55718[202.30.1.1:55718]-->202.10.1.2:20
48
```

从 Server-map 表中可以看到 NAT 类型是 No-PAT、NAT 转换前后的 IP 地址，由于端口没有转换，所以并没有显示端口信息。这里可以注意到正、反向 Server-map 表中的目的 IP 均为

any，也就是说只要 Server-map 表没有老化，理论上任何外网主机只要知道 NAT 转换后的 IP，都可以主动访问内网主机的公网 IP。

```
[SRG]display  firewall server-map

 server-map item(s)
 --------------------------------------------------------------------------------
 No-Pat, 192.168.0.2[202.30.1.1] -> any, Zone: ---
   Protocol: any(Appro: ---), Left-Time: 00:11:59, Addr-Pool: 1
   VPN: public -> public

 No-Pat Reverse, any -> 202.30.1.1[192.168.0.2], Zone: untrust
   Protocol: any(Appro: ---), Left-Time: --:--:--, Addr-Pool: ---
   VPN: public -> public
```

我们再从 PC1 上 ping Server 2，在 FW 再上查看会话表和 Server-map 表。各位发现了吗？做 NAT 转换后的公网 IP 还是 202.30.1.1，而不是 202.30.1.2。这说明：**做源 NAT 时，虽然选择哪个公网 IP 是随机的，但是这个公网 IP 由私网源 IP 决定，和目的 IP 无关**。只要私网源 IP 不变、地址池相关配置不变，同一个私网源 IP 就会固定的转换为同一个公网 IP。

```
<SRG>display firewall session table

 Current Total Sessions : 5
  icmp  VPN:public --> public 192.168.0.2:11930[202.30.1.1:11930]-->210.10.2.10:2
048
  icmp  VPN:public --> public 192.168.0.2:12186[202.30.1.1:12186]-->210.10.2.10:2
048
  icmp  VPN:public --> public 192.168.0.2:12442[202.30.1.1:12442]-->210.10.2.10:2
048
  icmp  VPN:public --> public 192.168.0.2:12698[202.30.1.1:12698]-->210.10.2.10:2
048
```

```
<SRG>display  firewall server-map

 server-map item(s)
 --------------------------------------------------------------------------------
 No-Pat, 192.168.0.2[202.30.1.1] -> any, Zone: ---
   Protocol: any(Appro: ---), Left-Time: 00:11:45, Addr-Pool: 1
   VPN: public -> public

 No-Pat Reverse, any -> 202.30.1.1[192.168.0.2], Zone: untrust
   Protocol: any(Appro: ---), Left-Time: --:--:--, Addr-Pool: ---
   VPN: public -> public
```

NAPT

NAPT 表示网络地址端口转换，即同时对 IP 地址和端口号进行转换，也可称为 PAT（PAT 不是只转换端口号的意思，而是 IP、端口号同时转换）。NAPT 是最常用的源 NAT 技术之一，他可以实现用少量公网 IP 满足大量私网用户上网的需求。

NAPT 和 NAT No-PAT 在配置上的区别仅在于选择不同的源 NAT 模式：NAPT 的 nat-policy 在指定 NAT 地址池时，不配置命令关键字“**no-pat**”，其他配置都是类似的。

例 5 NAPT 和 NAT No-PAT 配置上的差异点

```
#
nat-policy interzone trust untrust outbound
 policy 1
  address-group 1          //不配置 no-pat
```

从 PC1 上 ping PC2，在 FW 上查看会话表。可以看到源 IP 和源端口都做了 NAT 转换，而且端口号是顺序转换的。

```
[SRG]display  firewall session table

 Current Total Sessions : 5
  icmp  VPN:public --> public 192.168.0.2:43422[202.30.1.1:2048]-->202.10.1.2:204
8
  icmp  VPN:public --> public 192.168.0.2:43678[202.30.1.1:2049]-->202.10.1.2:204
8
  icmp  VPN:public --> public 192.168.0.2:43934[202.30.1.1:2050]-->202.10.1.2:204
8
  icmp  VPN:public --> public 192.168.0.2:44190[202.30.1.1:2051]-->202.10.1.2:204
8
  icmp  VPN:public --> public 192.168.0.2:44446[202.30.1.1:2052]-->202.10.1.2:204
8
```

再看 Server-map 表，没有显示信息？没错，NAPT 就是没有 Server-map 表！原因其实很好理解，NAPT 主要用于让大量用户上网，如果每个连接都建立 Server-map 表，则会占用大量的设备资源。

```
[SRG]display firewall server-map
```

从 PC1 上 ping Server 2，在 FW 上查看会话表。我们发现 NAPT 也是由源 IP 决定转换后的公网 IP，且端口顺序转换。端口从 2048 开始转换的现象说明：对于不同的连接来说， NAT 处理过程是彼此独立的。只要五元组不完全相同，就不用担心 NAT 转换冲突的问题（对于现在的网络通信来说，五元组完全一致的情况发生概率非常小）。

```
[SRG]display firewall session table

 Current Total Sessions : 5
  icmp  VPN:public --> public 192.168.0.2:2207[202.30.1.1:2048]-->210.10.2.10:204
8
  icmp  VPN:public --> public 192.168.0.2:2463[202.30.1.1:2049]-->210.10.2.10:204
8
  icmp  VPN:public --> public 192.168.0.2:2719[202.30.1.1:2050]-->210.10.2.10:204
8
  icmp  VPN:public --> public 192.168.0.2:2975[202.30.1.1:2051]-->210.10.2.10:204
8
  icmp  VPN:public --> public 192.168.0.2:3231[202.30.1.1:2052]-->210.10.2.10:204
8
```

出接口地址方式（easy-ip）

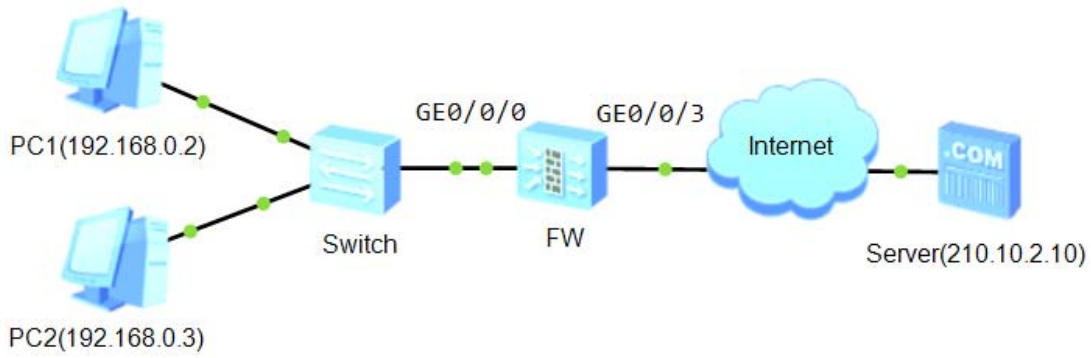
出接口地址方式是利用出接口的公网 IP 做源 NAT 转换，适用于公网 IP 非常少或接口动态获取 IP 的场景（**仅中低端防火墙支持接口动态获取 IP**）。

easy-ip 的 NAT 转换方式和 NAPT 一样，都是同时转换 IP 和端口。但是在具体配置上，高端

防火墙和中低端防火墙是不一样的：

- 高端防火墙需要配置 NAT 地址池，并将出接口 IP 配置在地址池中。实际上就是配置 NAPT 功能，只不过出接口 IP 和 NAT 地址池中的 IP 一样了。
- 中低端防火墙不需要配置 NAT 地址池，而是在 NAT 策略中指定做 easy-ip 转换。

我们使用如下组网进行演示。easy-ip 无需配置 NAT 地址池，只需在 NAT 策略中指定利用哪个接口做 easy-ip。安全策略的配置可以参考上面的 NAT No-PAT，easy-ip 不用配置黑洞路由。



例 6 配置 easy-ip

```
#
nat-policy interzone trust untrust outbound
  policy 1
    action source-nat
    source-address 192.168.0.0 0.0.0.255
    easy-ip GigabitEthernet0/0/3        //源 NAT 转换后的公网 IP 为接口 GE0/0/3 的 IP
```

分别从 PC1 和 PC2 上 ping Server，查看到会话表如下所示（部分 PC1 的会话已老化，GE0/0/3 的 IP 是 210.10.2.1/24）。可以看到源 IP 和端口都做了 NAT 转换，且转换后的端口是顺序增大的。这说明：虽然源 IP 不同，但是 NAT 转换都走同一个流程，所以端口号顺序增大。此外，和 NAPT 一样，easy-ip 也是没有 Server-map 表的。

```
[SRG]display  firewall session table

 Current Total Sessions : 8
  icmp  VPN:public --> public 192.168.0.2:65471[210.10.2.1:2050]-->210.10.2.10:20
48
  icmp  VPN:public --> public 192.168.0.2:192[210.10.2.1:2051]-->210.10.2.10:2048

  icmp  VPN:public --> public 192.168.0.2:704[210.10.2.1:2052]-->210.10.2.10:2048

  icmp  VPN:public --> public 192.168.0.3:7104[210.10.2.1:2053]-->210.10.2.10:204
8
  icmp  VPN:public --> public 192.168.0.3:7360[210.10.2.1:2054]-->210.10.2.10:204
8
  icmp  VPN:public --> public 192.168.0.3:7616[210.10.2.1:2055]-->210.10.2.10:204
8
  icmp  VPN:public --> public 192.168.0.3:7872[210.10.2.1:2056]-->210.10.2.10:204
8
  icmp  VPN:public --> public 192.168.0.3:8128[210.10.2.1:2057]-->210.10.2.10:204
8
```

今天我们先介绍这三种源 NAT，强叔会在下期继续和大家聊聊余下的两种源 NAT，敬请期待！

强叔提问

- 如果组网是多出口的情况（例如电信、联通），该如何配置地址池呢？
- 配置完 NAT No-PAT、NAPT 或 easy-ip 后，我们该如何检验是否配置成功？

小伙伴们对于这三种源 NAT 还有什么疑问，欢迎留言！

一墙当关，万夫上网——源 NAT（下篇）

大家好，强叔接着上回书继续和大家聊聊源 NAT。除了 NAT No-PAT、NAPT 和 easy-ip，华为防火墙还支持两种源 NAT 功能，如下所示：

- Smart NAT（仅高端防火墙 USG9000 系列支持）
- 三元组 NAT（仅高端防火墙 USG9000 系列支持）

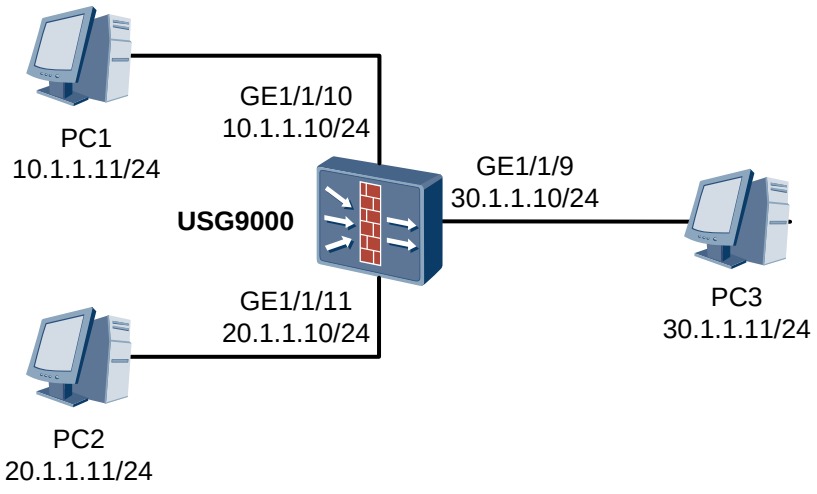
Smart NAT

Smart NAT 为何称为“聪明的 NAT”？这要从他和 NAT No-PAT、NAPT 的关联性说起：

我们假设 Smart NAT 使用的地址池中包含 N 个 IP，其中一个 IP 被指定为预留地址，另外 N-1 个地址构成地址段 1（section 1）。进行 NAT 转换时，Smart NAT 会先使用 section 1 做 NAT No-PAT 类型的转换，当 section 1 中的 IP 都被占用后，才使用预留 IP 做 NAPT 类型的转换。

其实 Smart NAT 可以理解为是对 NAT No-PAT 功能的增强，他防止了用户数量激增导致大量用户不能上网的情况，即克服了 NAT No-PAT 的缺点——只能让有限的用户上网，当用户数量大于地址池中 IP 数量时，后面的用户将无法上网，只能等待公网 IP 被释放（会话老化）。Smart NAT 预留一个公网 IP 做 NAPT 后，无论有多少新增用户需要上网，都能满足其需求。

由于 eNSP 不支持高端防火墙，所以我们通过一个实际的组网来看下 Smart NAT 的实现过程。



Smart NAT 的配置和 NAT No-PAT 的配置几乎完全一致，区别只是在地址池中指定了一个 IP 作为预留 IP，下面给出关键配置：

例 1 配置 Smart NAT

```
#
 nat address-group 1
  mode no-pat        //模式要选择 no-pat
  smart-nopat   30.1.1.21          //预留一个 IP 做 NAPT
  section 1   30.1.1.20   30.1.1.20        //section 中不能包含预留 IP!
#
policy interzone trust untrust outbound
 policy 1
  action permit
  policy source 10.1.1.0 0.0.0.255
  policy source 20.1.1.0 0.0.0.255
#
nat-policy interzone trust untrust outbound
 policy 1
  action source-nat
  address-group 1
  policy source 10.1.1.0 0.0.0.255
  policy source 20.1.1.0 0.0.0.255
```

先从 PC2 上 ping PC3，再从 PC1 上 ping PC3（请注意这个顺序），在 USG9000 上查看会话表（**中括号[]内的是 NAT 转换后的 IP 和端口**）。可以看到 PC2（20.1.1.11）只转换了 IP，没有转换端口，也就是说做了 NAT No-PAT 转换。而 PC1（10.1.1.11）的 IP 和端口都进行了转换，且转换后的 IP 就是预留 IP（30.1.1.21），所以说他做了 NAPT 转换。

```
[USG9000]display  firewall session table

Current total sessions: 2
Slot: 2 CPU: 3
icmp VPN: public --> public 10.1.1.11:44038[30.1.1.21:4096] --> 30.1.1.11:2048
icmp VPN: public --> public 20.1.1.11:44056[30.1.1.20:44056] --> 30.1.1.11:2048
```

NAPT

NAT No-PAT

我们再看 Server-map 表，结果也符合 NAT No-PAT 和 NAPT 功能的特点：只有 NAT No-PAT 类型的表项，NAPT 转换没有 Server-map 表。

```
[USG9000]display  firewall server-map
10:34:22  2014/04/25
 ServerMap item(s) on slot 2 cpu 3
 ------------------------------------------------------------------------------
 Type: No-Pat,  20.1.1.11[30.1.1.20] -> ANY,  Zone: untrust
 Protocol: ANY(Appro: unknown),  Left-Time:00:05:55,  Pool: 1, Section: 1
 Vpn: public -> public

 Type: No-Pat Reverse,  ANY -> 30.1.1.20[20.1.1.11],  Zone: untrust
 Protocol: ANY(Appro: unknown),  Left-Time:---,  Pool: 1, Section: 1
 Vpn: public -> public
```

三元组 NAT

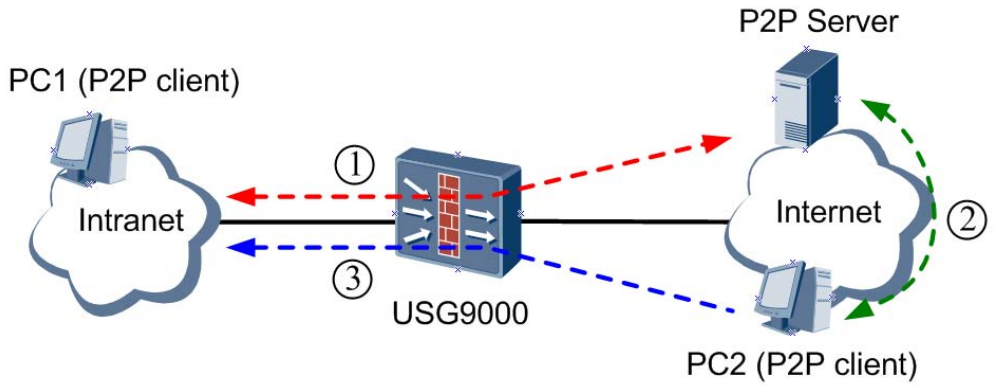
三元组 NAT 中的“三元”是指：源 IP、源端口和协议。

三元组 NAT 功能的产生基于这样一个事实：现今大部分网络主机都处在 NAT 设备后，而 P2P 应用（如 QQ、BT）是人们最常用的软件之一。当 NAT 遇到 P2P 的时候，产生的不是完美的“NAT-P2P”，而是……你可能下载不了 BT 资源、无法和女神聊 QQ 了。T_T

事实上，当有 NAT 设备存在时，如果要保证内网用户正常使用 P2P 软件，往往还需要其他网络设备来配合。而华为高端防火墙 USG9000 系列可以完美地解决这个问题：通过使用三元组 NAT 功能，防火墙可以在做 NAT 网关的同时支持 P2P 业务的正常交互，完全不需要其他设备！

为了引出三元组 NAT 的特点，我们先通过下图来看看 P2P 业务的一般交互流程。PC1 和 PC2 是两台运行 P2P 业务的客户端，他们运行 P2P 应用时首先会和 P2P 服务器进行交互（登录、认证等操作），服务器会记录客户端的地址和端口。当 PC2 需要下载文件时，服务器会将拥有该文件的客户端的地址和端口发送给 PC2（例如 PC1 的地址和端口），然后 PC2 会向 PC1 发送请求，并从 PC1 上下载文件。

如果 PC1 是内网主机，在防火墙上做 NAT 转换，此时 P2P 服务器记录的就是 PC1 做 NAT 转换后的地址和端口。也许有人会问：PC1 做 NAT 转换后的地址和端口不会变化吗？答案是会变化，但是 PC1 会定期向服务器发送报文（用于认证等），服务器也就记录了最新的 NAT 后地址和端口，所以可以保证 PC2 能够成功访问 PC1。



上述过程看起来似乎没有问题，但是对于防火墙来说，这里有两个问题：

- PC1 没有主动访问过 PC2，一般来说，防火墙不会允许 PC2 主动访问 PC1。
- PC1 访问服务器时，NAT 后的地址和端口只能被服务器用来访问 PC1，其他主机（例如

PC2）不能利用这个地址和端口访问 PC1，请求报文在防火墙上将被丢弃。

三元组 NAT 可以完美地解决上述两个问题，依靠的正是以下两个特点：

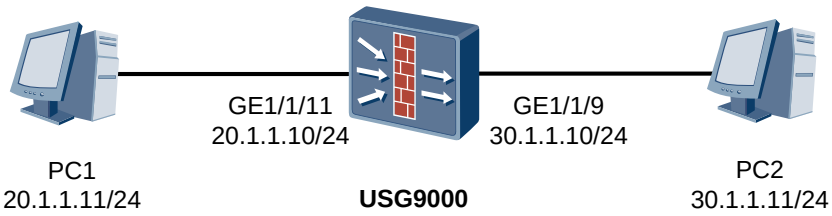
- 支持外网主动访问

无论内网主机是否主动访问过某个外网主机，只要外网主机知道内网主机 NAT 转换后的地址和端口，就可以主动向该内网主机发起访问。

- 动态对外端口一致性

内网主机做 NAT 转换后的地址和端口将在一段时间内保持不变，在此时间内段，内网主机固定地使用此 NAT 后地址和端口访问任意外网主机，任意外网主机也可以通过此 NAT 后地址和端口访问内网主机。

从实现原理角度讲，三元组 NAT 是通过 Server-map 表使外网主机可以主动访问内网主机，并保证 NAT 转换关系在一段时间内保持不变。下面通过一个例子来说明会更容易理解（我们还是用实际设备来组网）。



三元组 NAT 和 NAT No-PAT 在配置上的区别仅在于选择不同的源 NAT 模式，下面给出关键配置：

例 2 配置三元组 NAT

```
#
 nat address-group 1
   mode full-cone global //做三元组 NAT 转换，global 表示 Server-map 表不限制域间关系
   section 1 30.1.1.20 30.1.1.20
#
nat-policy interzone trust untrust outbound
 policy 1
   action source-nat
   address-group 1
   policy source 20.1.1.0 0.0.0.255
```

从 PC1 上 ping PC2，在会话表中可以看到源 IP 和端口都做了 NAT 转换。

```
<USG9000>display  firewall session table

Current total sessions: 2
Slot: 2 CPU: 3
icmp VPN: public --> public 20.1.1.11:44092[30.1.1.20:3536] --> 30.1.1.11:2048
```

再看 Server-map 表，可以看到类型是 FullCone（全圆锥），即三元组 NAT 的 Serverv-map 表（关于 FullCone 的内容，请见下面的附录内容）。在 Server-map 表没有老化前，三元组的源 Server-map 表项（FullCone Src）作用是：内网主机访问任意外网主机（ANY）时，NAT 转换后的地址和端口都是 30.1.1.20:3536。目的 Server-map 表项（FullCone Dst）的作用是：任意外网主机（ANY）都可以通过 30.1.1.20:3536 来访问内网主机 20.1.1.11。通过这两条 Server-map 表项即实现了“外网主机可以主动访问内网主机，并保证 NAT 转换关系在一段时间内保持不变”的要求。

```
<USG9000>display  firewall server-map

ServerMap item(s) on slot 2 cpu 3
------------------------------------------------------------------------------
Type: FullCone Src,  20.1.1.11:44092[30.1.1.20:3536] -> ANY,  Zone:---
Protocol: icmp(Appro: ---),  Left-Time:00:00:58,  Pool: 1, Section: 0
Vpn: public -> public
Hotversion: 2

Type: FullCone Dst,  ANY -> 30.1.1.20:3536[20.1.1.11:44092],  Zone:---
Protocol: icmp(Appro: ---),  Left-Time:00:00:58,  Pool: 1, Section: 0
Vpn: public -> public
Hotversion: 2
```

由于实验室环境限制，此处无法通过 P2P 应用来验证三元组 NAT 的另一个特点：PC1 没有访问过的主机可以通过 30.1.1.20:3536 主动访问 PC1。实际上，如果有一台 P2P 客户端 PC3，他是可以通过 30.1.1.20:3536 成功访问 PC1 的。通过另一种方法可以验证 NAT 转换的固定性：在 Serverv-map 表没有老化前，可以用一台主机 telnet 30.1.1.20:3536，当然 PC1 要先开启 Telnet 服务器功能，且端口要设置为 3536。

至此，华为防火墙支持的源 NAT 功能都介绍完了，下次我们将介绍目的 NAT 相关内容，敬请期待！

强叔提问

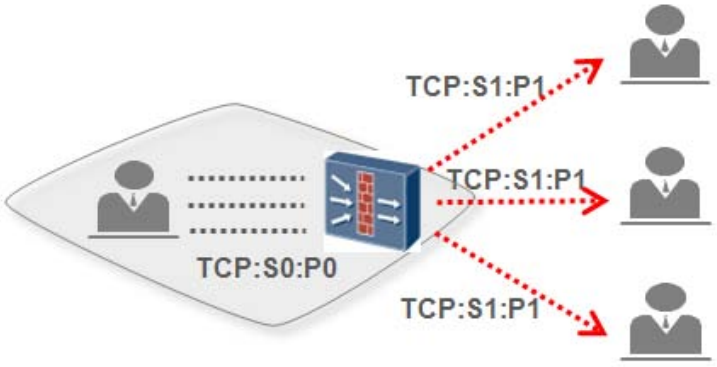
- 小伙伴们，你们还记得有哪几类源 NAT 功能会生成 Server-map 表吗？
- 各种源 NAT 功能都应用在什么场景下？

大家对源 NAT 还有什么疑问？欢迎留言~

附录

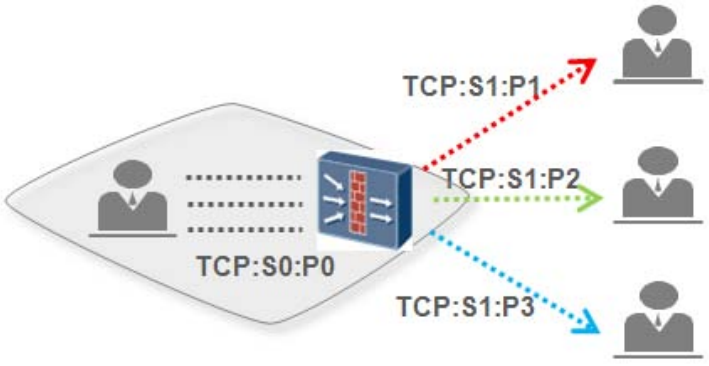
根据 RFC3489 中的内容，Full Cone（全圆锥）是 4 种 NAT 端口映射方式中的一种，其他 3 种分别为：Restricted Cone（受限圆锥）、Port Restricted Cone（端口受限圆锥）和 Symmetric（对称型）。

- 全圆锥 NAT 的模型是：内网主机做 NAT 转换后的地址和端口在一段时间内保持不变，不会因为目的地址不同而不同，所以内网主机可以使用相同的 NAT 后三元组（源 IP、源端口、协议）访问不同外网主机。当 NAT 后三元组确定后，外网主机也都可以通过该三元组访问内网用户。（题外话：之所以叫“全圆锥”，应该就是根据“一对多”的模型命名的吧。）



- 对称型 NAT 的模型是：内网主机会根据不同的目的地址做 NAT 转换，NAT 转换后的地址和端口一般是不相同的。由于对不同外网主机呈现不同的三元组（源 IP、源端口、协议），所以外网主机只能通过访问对应的 NAT 后三元组才能进入内网，即需要限定目标用户和端口，因此对称型 NAT 也称为五元组 NAT（源 IP、源端口、目的 IP、目的端口、协议）。

华为防火墙除了支持全圆锥 NAT，也支持对称型 NAT，NAPT 功能即为五元组 NAT。



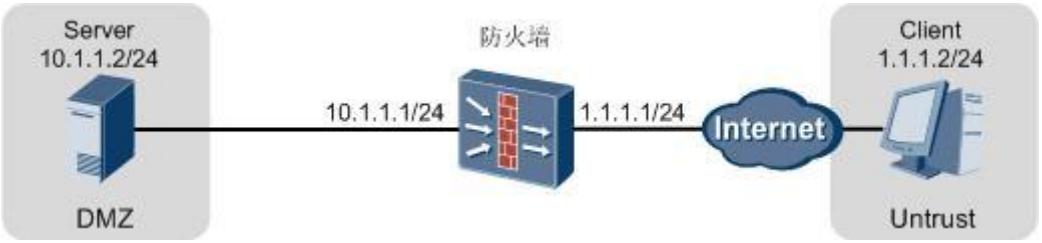
NAT Server 基础

学校或公司的私网通常会有一些服务器需要提供给公网用户访问。但是网络部署时，服务器地址一般都会被配置成私网地址，这样服务器就不能直接使用自身的地址来提供服务了。那么，防火墙作为学校或企业的出口网关时，是如何应对这个问题的呢？

如果小伙伴们有读过强叔的源 NAT 篇，聪明的你一定会想到，防火墙是不是也可以将服务器的私网地址通过 NAT 转换成公网地址来提供服务呢？

Bingo！你的大方向已经对了。不过，源 NAT 是对私网用户访问公网的报文的源地址进行转换，而服务器对公网提供服务时，是公网用户向私网发起访问，方向正好反过来了。于是，NAT 转换的目标也由报文的源地址变成了目的地址。针对服务器的地址转换，我们赋予了它一个形象的名字——NAT Server（服务器映射）。

下面来看下防火墙上的 NAT Server 是如何配置和实现的。



在防火墙上配置如下命令，就能将上图中服务器的私网地址 10.1.1.2 映射成公网地址 1.1.1.1。

```
[FW] nat server global 1.1.1.1 inside 10.1.1.2
```

但是，如果一台服务器同时存在多种协议和端口的服务项，按照上述配置会将服务器上所有服务项都发布到公网，这无疑会带来很大的安全风险。华为防火墙支持配置指定协议的 NAT Server，只将服务器上特定的服务项对公网发布，从而避免服务项全发布带来的风险。例如，我们可以按如下方式配置，将服务器上 80 端口的服务项映射为 9980 端口供公网用户访问。

```
[FW] nat server protocol tcp global 1.1.1.1 9980 inside 10.1.1.2 80
```

这里将 80 端口转换为 9980 端口而不是直接转换成 80 端口是因为，一些地区的运营商会阻断新增的 80、8000、8080 端口的业务，从而导致服务器无法访问。

小伙伴们是否还记得《*安全策略篇 ASPF：隐形通道*》中提到的 Server-map 表，NAT server 配置完成之后，也会生成 Server-map 表来保存映射关系。不过与 ASPF Server-map 表项的

动态老化不同的是，NAT Server 的 Server-map 表项是静态的，只有当 NAT Server 配置被删除时，对应的 Server-map 表项才会被删除。

```
[FW]display firewall server-map
07:18:56  2014/05/11
 server-map item(s)
 --------------------------------------------------------------------------------
 Nat Server, any -> 1.1.1.1:9980[10.1.1.2:80], Zone: ---
   Protocol: tcp(Appro: unknown), Left-Time: --:--:--, Addr-Pool: ---
   VPN: public -> public
```

上图就是 NAT Server 的 Server-map 表项。图中红框标注的字段就记录着服务器私网地址端口和公网地址端口的映射关系。[]内为服务器私网地址和端口、[]外为服务器公网地址和端口。我们将表项翻译成文字就是：任意客户端（any）向（->）1.1.1.1:9980 发起访问时，报文的目的地址和端口都会被转换成 10.1.1.2:80。具体的流程如下：

当客户端通过 1.1.1.1:9980 访问服务器时，防火墙收到报文的首包后，首先是查找并匹配到 Server-map 表项，将报文的目的地址和端口转换为 10.1.1.2:80。然后根据目的地址判断出报文在哪两个安全区域间流动，报文通过域间安全策略检查后，防火墙会建立如下的会话表，并将报文转发到私网。

```
[FW]display firewall session table
07:25:33  2014/05/11
 Current Total Sessions : 1
  http  VPN:public --> public 1.1.1.2:2049-->1.1.1.1:9980[10.1.1.2:80]
```

之后，服务器对客户端的请求做出响应。响应报文到达防火墙后匹配到上面的会话表，防火墙将报文的源地址和端口转换为 1.1.1.1:9980，而后发送至公网。后续客户端继续发送给服务器的报文，防火墙都会直接根据会话表对其进行地址和端口转换，而不会再去查找 Server-map 表项了。

在防火墙的前后抓包，能很清楚地看到 NAT Server 的效果：

- 转换客户端发往服务器的报文的目的地址和端口。

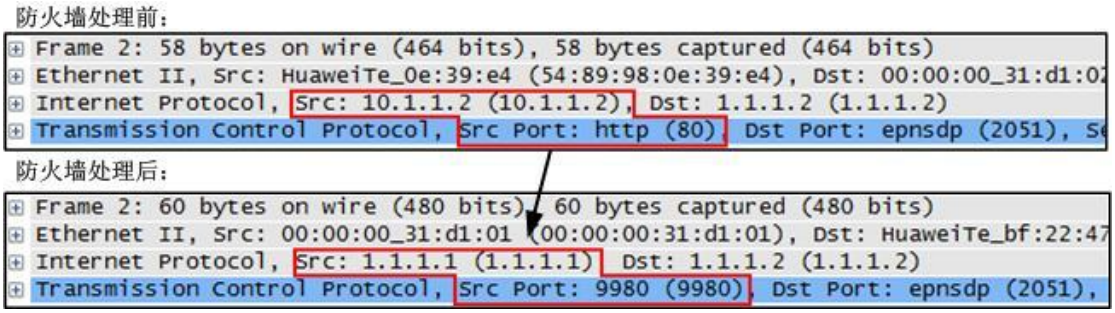
防火墙处理前：

```
⊞ Frame 1: 58 bytes on wire (464 bits), 58 bytes captured (464 bits)
⊞ Ethernet II, Src: HuaweiTe_bf:22:47 (54:89:98:bf:22:47), Dst: 00:00:00_31:d1:0
⊞ Internet Protocol, Src: 1.1.1.2 (1.1.1.2), Dst: 1.1.1.1 (1.1.1.1)
⊞ Transmission Control Protocol, Src Port: epnsdp (2051), Dst Port: 9980 (9980),
```

防火墙处理后：

```
⊞ Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
⊞ Ethernet II, Src: 00:00:00_31:d1:02 (00:00:00:31:d1:02), Dst: HuaweiTe_0e:39:e
⊞ Internet Protocol, Src: 1.1.1.2 (1.1.1.2), Dst: 10.1.1.2 (10.1.1.2)
⊞ Transmission Control Protocol, Src Port: epnsdp (2051), Dst Port: http (80), S
```

- 转换服务器响应客户端的报文的源地址和端口。



以上就是防火墙 NAT Server 的基本配置和工作原理，通过强叔的讲解，小伙伴们想必已掌握其中的奥妙了吧。

但这些仅仅是 NAT Server 的基础知识，知道这些只能算是初窥门径。要想成为能从容应对现网中各种 NAT Server 配置的顶级高手，小伙伴们还需要研习并掌握强叔自创的 NAT Server 三十二字真言：

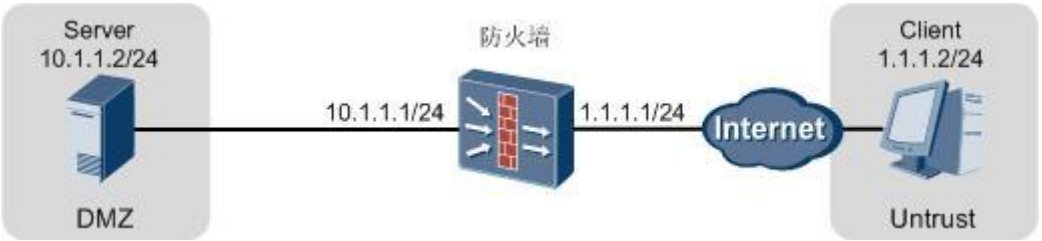
一正一反，出入自如　去反存正，自断出路

一分为二，源进源回　虚实变换，合二为一

接下来的两期中强叔将会向小伙伴们阐释这三十二字真言的内涵所在，敬请期待。

强叔提问

本篇开头的组网中，强叔只给出了 NAT Server 的配置，那安全策略如何配置呢？

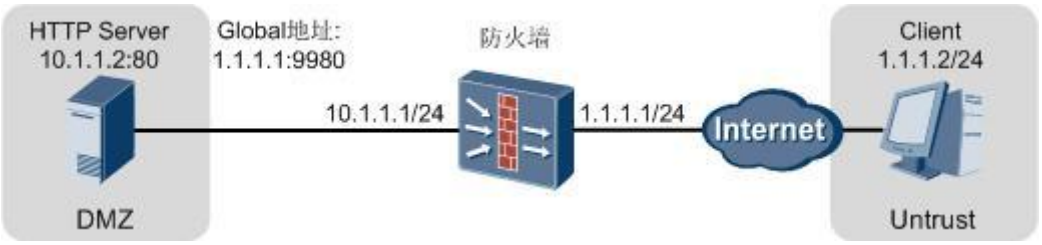


NAT Server 三十二字真言（上篇）

NAT Server 基础篇放出来后，论坛的小伙伴大呼不过瘾。为了感谢大家的支持，强叔挑灯夜战，总算是把三十二字真言的前半段内容给赶制出来了。还记得前面十六个字的内容吗？——一正一反，出入自如；去反存正，自断出路。

一正一反，出入自如

所谓入，是指公网用户访问私网服务器；所谓出，是指私网服务器主动访问公网。下面强叔就要向大家展示下防火墙配置 NAT Server 后，如何做到公网用户和私网服务器之间的出入自如。以下内容继续围绕基础篇中的组网和配置来展开。




```
[FW] nat server protocol tcp global 1.1.1.1 9980 inside 10.1.1.2 80
```

在基础篇中强叔展示给大家的 Server-map 表项其实还隐藏了一部分，完整的表项应该是这样的：

```
<FW>display firewall server-map
08:11:02  2014/05/11
 server-map item(s)
 ------------------------------------------------------------------------------
Nat Server, any -> 1.1.1.1:9980[10.1.1.2:80], Zone: ---
   Protocol: tcp(Appro: unknown), Left-Time: --:--:--, Addr-Pool: ---
   VPN: public -> public

Nat Server Reverse, 10.1.1.2[1.1.1.1] -> any, Zone: ---
   Protocol: any(Appro: ---), Left-Time: --:--:--, Addr-Pool: ---
   VPN: public -> public
```

Nat Server, any -> 1.1.1.1:9980[10.1.1.2:80]为正向 Server-map 表项，其作用为入。在公网用户访问服务器时对报文的目的地址做转换。

Nat Server Reverse, 10.1.1.2[1.1.1.1] -> any 为反向 Server-map 表项，其作用为出。当私网服务器主动访问公网时，可以直接使用这个表项将报文的源地址由私网地址转换为公网地址，而不用再单独为服务器配置源 NAT 策略。这就是防火墙 NAT Server 做的非常贴心的地方了，

一条命令同时打通了私网服务器和公网之间出入两个方向的**地址转换**通道。

请注意强叔此处的用词，通道前面加上了“地址转换”四个字。没错，不论是正向还是反向 Server-map 表项，都仅能实现地址转换而已，并不能像 ASPF 的 Server-map 表项一样打开一个可以绕过安全策略检查的临时通道。因此，公网用户要能访问私网服务器或者服务器要能访问公网，还需要配置正确的域间安全策略。

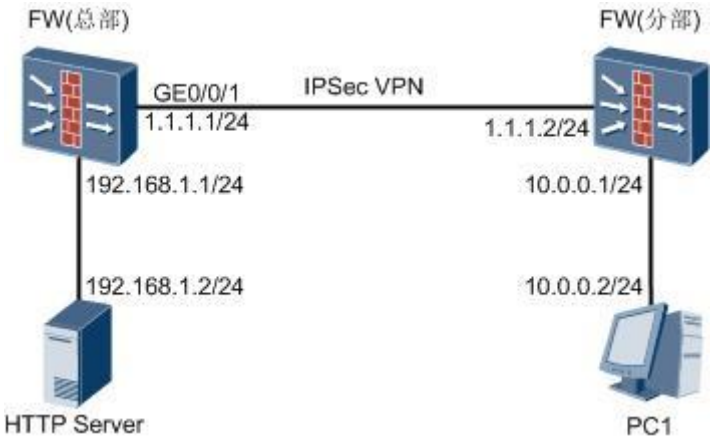
去反存正，自断出路

顾名思义，去反存正就是删除反向 Server-map 表项。配置 NAT Server 时带上 no-reverse 参数就能让生成的 Server-map 表项只有正向没有反向。

```
[FW] nat server protocol tcp global 1.1.1.1 9980 inside 10.1.1.2 80 no-reverse
```

```
<FW>display firewall server-map
08:27:40  2014/05/11
 server-map item(s)
 --------------------------------------------------------------------------------
 Nat Server, any -> 1.1.1.1:9980[10.1.1.2:80], Zone: ---
   Protocol: tcp(Appro: unknown), Left-Time: --:--:--, Addr-Pool: ---
   VPN: public -> public

<FW>
```

没有了反向 Server-map 表项，也就相当于断去了服务器到公网的出路。那何时需要自断出路呢？首先，让我们来看看下面这个案例。



上图中总部有一台服务器需要提供给公网用户访问，于是在总部防火墙上配置了如下的 NAT Server：

```
[FW] nat server protocol tcp global 1.1.1.1 9980 inside 192.168.1.2 80
```

同时，总部和分支之间通过 IPSec VPN 实现互访。总部防火墙 IPSec 的部分配置如下：

```
#
```

```
acl number 3000          //定义需要进行 IPSec 封装的数据流
  rule 5 permit ip source 192.168.1.0 0.0.0.255 destination 10.0.0.0 0.0.0.255
#
ipsec policy map1 10 manual
 security acl 3000       //引用 acl，只有符合 acl3000 的数据流才会被送入 IPSec 隧道封装
 proposal tran1
 …
```

因为总部 192.168.1.0/24 网段员工需要访问公网，所以还配置了如下的源 NAT 策略：

```
#
nat-policy interzone trust untrust outbound
  policy 5
  action source-nat
  policy source 192.168.1.0 mask 24
  easy-ip GigabitEthernet0/0/1
```

不过仅配置这条源 NAT 策略是不够的。因为这条源 NAT 策略会将 trust 区域中 192.168.1.0/24 网段发往 untrust 区域的所有报文的源地址都转换成 GE0/0/1 接口的地址 1.1.1.1。熟悉 IPSec 的小伙伴们应该知道，报文源地址如果变成了 1.1.1.1，就不会匹配到 ACL 3000，也就不会进入 IPSec 隧道进行封装，这样总部就别想通过 IPSec VPN 和分部之间通信了。所以，除了上面这条源 NAT 策略，还需要配置一条对总部访问分部的流量不做源地址转换的 NAT 策略，具体如下：

```
#
nat-policy interzone trust untrust outbound
  policy 0
  action no-nat
  policy source 192.168.1.0 mask 24
  policy destination 10.0.0.0 mask 24
```

注意

上面两条源 NAT 策略，policy0 的匹配条件要比 policy5 更加严格，所以配置完成后需要确认策略列表中 policy0 在 policy5 之上。否则报文匹配到条件宽松的 policy5 后就直接做了源地址转换，而不会再匹配到 policy0 了。

配置完成后，我们发现了一个很奇怪的现象：分部的员工可以访问总部服务器的私网地址 192.168.1.2，总部 192.168.1.0/24 网段的员工也能正常和分部的 10.0.0.0/24 网段通信。但总部的服务器却无法访问分部 10.0.0.0/24 网段的资源，删除 NAT Server 配置后就能正常访问。

很明显，问题就出在 NAT Server 上，但因为总部的服务器需要提供给公网用户访问，我们不能随意将 NAT Server 配置去掉，那该如何解决这个问题呢？下面就让强叔来给小伙伴们分析下根因所在并给出解决办法。

总部 Server ping 分部 PC 时，总部的防火墙上可以看到这样一条会话：

```
<FW> display firewall session table source inside 192.168.1.2
  icmp  VPN:public --> public 192.168.1.2:512[1.1.1.1:512]-->10.0.0.2:2048
```

可以看出，防火墙将报文的源地址由 192.168.1.2 转变成了 1.1.1.1。但我们明明已经配置了一条对 192.168.1.0/24 网段发往 10.0.0.0/24 网段的报文不做源地址转换的 NAT 策略啊，为什么源地址还是被转换了呢？

我们先使用 **display nat-policy all** 命令来查看和确认下源 NAT 策略的命中情况：

```
#
nat-policy interzone trust untrust outbound
 policy 0 (0 times matched)
  action no-nat
  policy service service-set ip
  policy source 192.168.1.0 mask 24
  policy destination 10.0.0.0 mask 24
```

结果显示，确实没有报文命中源 NAT 策略。

接下来，让我们取消 NAT Server 的配置，再次从总部 Server ping 分部的 PC1，并查看源 NAT 策略的命中情况。这时你会发现，有报文命中源 NAT 策略了！

```
#
nat-policy interzone trust untrust outbound
 policy 0 (1 times matched)
  action no-nat
  policy service service-set ip
  policy source 192.168.1.0 mask 24
  policy destination 10.0.0.0 mask 24
```

所以，肯定是配置 NAT Server 时引入的什么东东先把地址给转换了，导致匹配不到源 NAT 策略。说到这里，再联想下真言的前八个字，相信小伙伴们已经知道问题所在了吧。没错，幕后的“黑手”正是 NAT Server 生成反向 Server-map 表项：

```
Nat Server Reverse, 192.168.1.2[1.1.1.1] -> any, Zone: ---
```

防火墙的报文处理流程中，反向 Server-map 表是比源 NAT 策略优先匹配的，报文匹配到反向 Server-map 表后，就不会再匹配源 NAT 策略了。这样，最终生成的会话表中依然有一个源地址转换的信息(也就是反向 Server-map 表中的源地址转换信息)，报文在进入 IPSec VPN

隧道封装之前，设备还是会根据会话表将报文的源地址转换为 1.1.1.1。

找到问题的根因所在，解决办法也就有了。配置 NAT Server 时加上 no-reverse 参数，不生成反向 Server-map 表项就可以了。

```
[FW] nat server protocol tcp global 1.1.1.1 9980 inside 192.168.1.2 80 no-reverse
```

当然，no-reverse 参数不仅限于这个场景中使用，后面我们还会提到它。小伙伴们只需要记住配置了这个参数后，就不会生成反向 Server-map 表项这一结论，再根据遇到的具体问题灵活运用即可。

以上就是三十二字真言前半段的所有内容了，不知小伙伴们是否领悟到了其中的真谛？下一期强叔将会向大家解释后半段的含义，敬请期待。

强叔提问

配置 NAT Server 后 ASPF 的 Server-map 表项会有什么变化呢？小伙伴们可以在自己的设备或者模拟器上实验对比下。

NAT Server 三十二字真言（下篇）

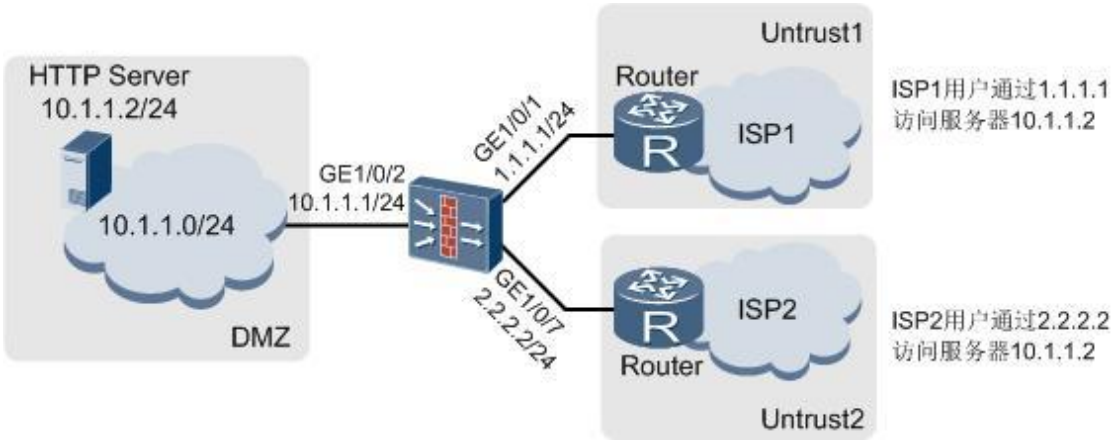
本期强叔将会给大家阐释三十二字真言的后半段：一分为二，源进源回；虚实变换，合二为一。

一分为二，源进源回

防火墙作为出口网关，双出口、双 ISP 接入公网时，配置 NAT Server 通常需要一分为二，让一个私网服务器向两个 ISP 发布两个不同的公网地址供访问。一分为二的方法有两种：

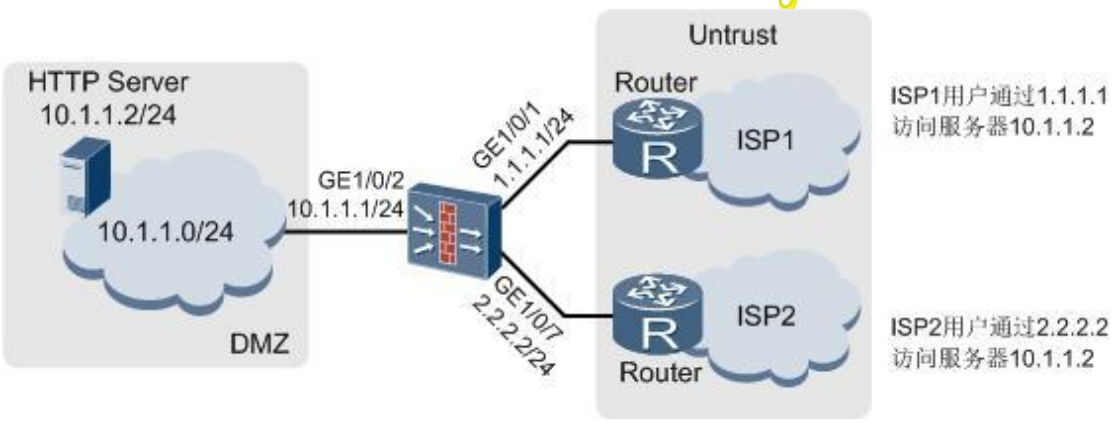
第一种是将接入不同 ISP 的公网接口规划在不同的安全区域中，配置 NAT Server 时，带上 zone 参数，使同一个服务器向不同安全区域发布不同的公网地址。

```
[FW] nat server zone untrust1 protocol tcp global 1.1.1.1 9980 inside 10.1.1.2 80
[FW] nat server zone untrust2 protocol tcp global 2.2.2.2 9980 inside 10.1.1.2 80
```



第二种是将接入不同 ISP 的公网接口规划在同一个安全区域中，配置 NAT Server 时，带上 no-reverse 参数，使同一个服务器向同一个安全区域发布两个不同的公网地址。

```
[FW] nat server protocol tcp global 1.1.1.1 9980 inside 10.1.1.2 80 no-reverse
[FW] nat server protocol tcp global 2.2.2.2 9980 inside 10.1.1.2 80 no-reverse
```



看到这里小伙伴们就要问了，强叔强叔，上一期中你不是讲过 no-reverse 参数是用来除去反向 Server-map 表项自断出路的吗，这里怎么又用到了呢？莫急莫急，且听强叔给你慢慢道来。

首先，我们来看下不带 no-reverse 参数直接配置上面两条命令会发生什么？

答案是不带 no-reverse 参数这两条命令压根就不能同时下发。

```
[FW]nat server protocol tcp global 1.1.1.1 9980 inside 10.1.1.2 80
09:31:33  2014/05/11
[FW]nat server protocol tcp global 2.2.2.2 9980 inside 10.1.1.2 80
09:31:45  2014/05/11
  错误：该内部地址已经被使用！
```

我们再尝试着逆向思考下，假如这两条命令能同时下发，会发生什么？

将上面的两条命令分别在两台防火墙上配置，然后查看各自生成的 Server-map 表项。

```
Nat Server, any -> 1.1.1.1:9980[10.1.1.2:80], Zone: ---
  Protocol: tcp(Appro: unknown), Left-Time: --:--:--, Addr-Pool: ---
  VPN: public -> public

Nat Server Reverse, 10.1.1.2[1.1.1.1] -> any, Zone: ---
  Protocol: any(Appro: ---), Left-Time: --:--:--, Addr-Pool: ---
  VPN: public -> public
```

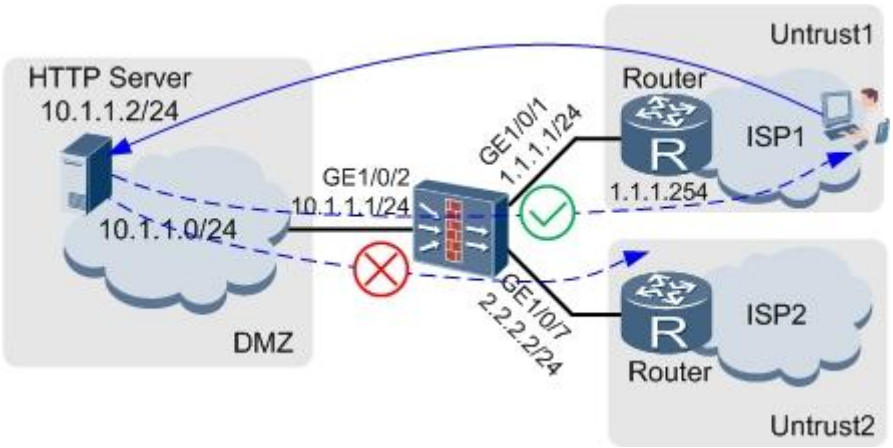
```
Nat Server, any -> 2.2.2.2:9980[10.1.1.2:80], Zone: ---
  Protocol: tcp(Appro: unknown), Left-Time: --:--:--, Addr-Pool: ---
  VPN: public -> public

Nat Server Reverse, 10.1.1.2[2.2.2.2] -> any, Zone: ---
  Protocol: any(Appro: ---), Left-Time: --:--:--, Addr-Pool: ---
  VPN: public -> public
```

很容易看出来，一台防火墙上的反向 Server-map 表项是将报文的源地址由 10.1.1.2 转换为 1.1.1.1，另一台防火墙上的反向 Server-map 表项是将报文的源地址由 10.1.1.2 转换为 2.2.2.2。试想下，如果这两个反向 Server-map 表项同时出现在一台防火墙上会发生什么？——防火

墙既可以将报文的源地址由 10.1.1.2 转换为 1.1.1.1，又可以转换为 2.2.2.2。于是乎，防火墙凌乱了~这就是两条命令不带 no-reverse 参数同时下发会带来的问题。如果配置时带上 no-reverse 参数，就不会生成反向 Server-map 表项。没有了反向 Server-map 表项，上述的问题也就不复存在了。

此外，一分为二时还会存在报文来回路径不一致的问题。例如，公网用户通过防火墙发布给 ISP1 的公网地址 1.1.1.1 访问服务器，服务器的响应报文到达防火墙后，防火墙根据目的地址查找路由表，可能会将响应报文由 ISP2 发送出去，这样就会导致访问速度过慢或无法访问。

为了避免这个问题，还需要在防火墙上增加一些的配置，保证报文的源进源回，即请求报文从某条路径进入，响应报文依然沿着同样的路径返回。



USG9000 系列防火墙源进源回功能是通过在公网接口下配置 **redirect-reverse** 命令来实现的。例如上图中接入 IPS1 的公网接口 GE1/0/1 的源进源回功能配置如下：

```
[FW] interface GigabitEthernet 1/0/1
[FW-GigabitEthernet1/0/1] redirect-reverse next-hop 1.1.1.254
```

配置完成后，如果请求报文从 GE1/0/1 进入，则响应报文也强制从 GE1/0/1 发出，而不再是通过查找路由表来确定出接口。

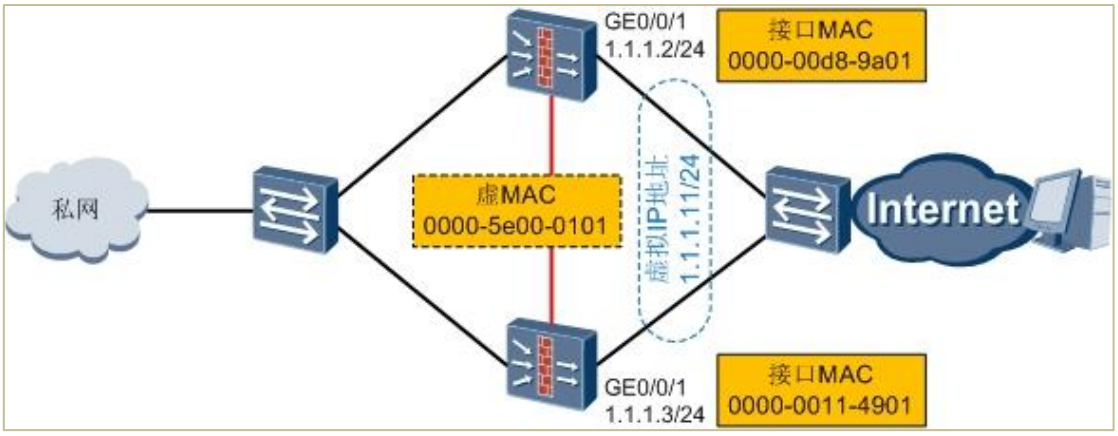
USG2000/5000/6000 系列防火墙源进源回功能配置思路与 USG9000 系列相同，配置命令为 **reverse-route next-hop** *next-hop-address*。

虚实变换，合二为一

为了让小伙伴们能明白“虚实”二字的含义，需要大家随着强叔穿越到未来的双机热备站，

提前了解一点双机热备的知识。

如下图所示的双机热备组网中，两台防火墙并不是直接使用 GE0/0/1 接口的实 IP 地址与公网通信，而是将 GE0/0/1 接口加入一个 VRRP 备份组，使用 VRRP 备份组的虚拟 IP 地址与公网通信。配置虚拟 IP 地址的同时，防火墙会自动为其生成一个虚 MAC 地址。



让我们再回到 NAT 站，强叔这里所说的 “实”指的就是物理接口的实 MAC 地址，“虚”指的就是虚 MAC 地址。

明白“虚实”的含义后，接下来强叔就要讲讲“虚实”在对 NAT Server 配置的影响。

首先，小伙伴们需要知道这样一个结论：当 NAT Server 公网地址与公网接口的地址在同一个网段时，防火墙会发送 NAT Server 公网地址的免费 ARP 请求报文。我们使用如下组网进行演示：

NAT Server 的配置如下：

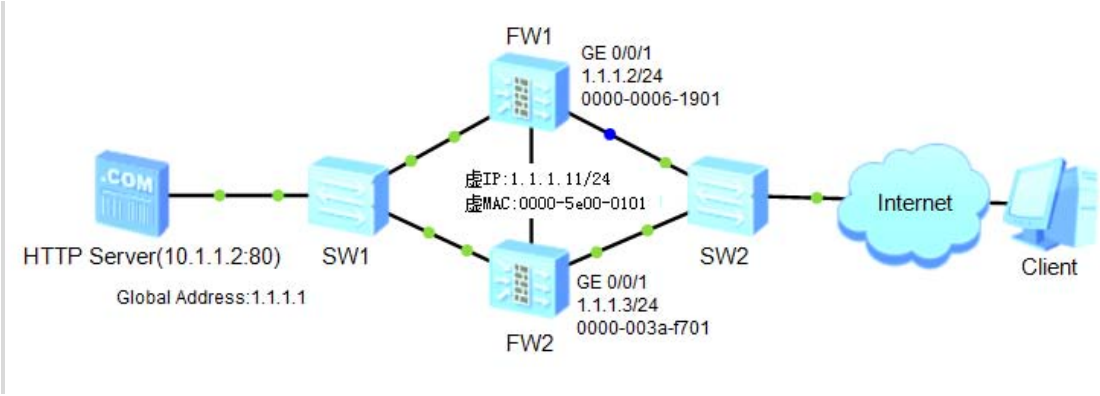
```
[FW] nat server protocol tcp global 1.1.1.1 9980 inside 10.1.1.2 80
```

在图示处抓包，可以看到防火墙发送的 NAT Server 公网地址的免费 ARP 请求报文。报文中携带的 1.1.1.1 的 MAC 地址为 0000-00e0-bb01，正是公网接口 GE0/0/1 的 MAC 地址。

```
Frame 1: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
Ethernet II, Src: 00:00:00_e0:bb:01 (00:00:00:e0:bb:01), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
Address Resolution Protocol (request/gratuitous ARP)
    Hardware type: Ethernet (0x0001)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (0x0001)
    [Is gratuitous: True]
    Sender MAC address: 00:00:00_e0:bb:01 (00:00:00:e0:bb:01)
    Sender IP address: 1.1.1.1 (1.1.1.1)
    Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
    Target IP address: 1.1.1.1 (1.1.1.1)
```

我们在 eNSP 上模拟了前面的双机热备组网，并在 FW1（主设备）上配置了 NAT Server：

```
[FW1] nat server protocol tcp global 1.1.1.1 9980 inside 10.1.1.2 80
```



命令一下发，设备就打印如下的 IP 地址冲突日志：

```
2014-05-11 10:02:22 FW1 %%01ARP/4/DUP_IPADDR(l): Receive an ARP packet with duplicate ip address 1.1.1.1 from GigabitEthernet0/0/1, source MAC is 0000-003a-f701!
```

日志中显示冲突源的 MAC 地址为 0000-003a-f701，这个正是 FW2 的 GE0/0/1 接口的 MAC。稍作分析，就能明白为什么会发生 IP 地址冲突了。

FW1 上配置了 NAT Server 后，由于公网地址为 1.1.1.1，和 GE0/0/1 接口的地址（1.1.1.2/24）在同一个网段，FW1 会发送 1.1.1.1 的免费 ARP 请求报文。报文中携带的 1.1.1.1 的 MAC 地址为 GE0/0/1 接口的 MAC 0000-0006-1901。同时，因为 FW1 和 FW2 处于双机热备状态，FW1 上 NAT Server 的配置会同步到 FW2 上，而 FW2 GE0/0/1 接口的地址（1.1.1.3/24）和 NAT Server 公网地址也在同一个网段，这样 FW2 也会发送 1.1.1.1 的免费 ARP 请求报文。报文中携带的 1.1.1.1 的 MAC 地址为 GE0/0/1 接口的 MAC 0000-003a-f701。于是，同一广播域中有两个 MAC 地址对应着同一个 IP 地址 1.1.1.1，产生了 IP 地址冲突。

同时，由于 FW1 和 FW2 同时发送免费 ARP 请求报文，上行设备学习到的 1.1.1.1 的 MAC 也会在 0000-0006-1901 和 0000-003a-f701 之间不停的切换。如下图就是 Client 上查看到的 ARP 表项。

```
PC>arp -a

Internet Address      Physical Address      Type
1.1.1.1               00-00-00-06-19-01     dynamic

PC>arp -a

Internet Address      Physical Address      Type
1.1.1.1               00-00-00-3A-F7-01     dynamic
```

这样，从 Client 上访问 1.1.1.1 时，Client 的网卡会时而用 0000-0006-1901 来封装报文，时而用 0000-003a-f701 来封装报文。如果用 0000-0006-1901 来封装报文，则报文会被发往 FW1（主设备），业务访问正常。如果用 0000-003a-f701 来封装报文，则报文会被发往 FW2（备设备）。由于 FW2 作为备设备时是不处理业务的，报文到达 FW2 后就会被丢弃。于是就会出现业务时通时不通的情况。

在配置命令中加上 vrrp 关键字就能解决这个问题。我们按如下命令重新配置：

```
[FW1] nat server protocol tcp global 1.1.1.1 9980 inside 10.1.1.2 80 vrrp 1
```

首先，设备上不再打印 IP 地址冲突日志了。在防火墙和上行交换机之间抓包我们会发现，只有主用防火墙会发送免费 ARP 报文，且报文中携带的 1.1.1.1 的 MAC 地址变成了 0000-5e00-0101，VRRP 备份组 1 的虚 MAC 地址。Client 访问 1.1.1.1 时，网卡会使用 0000-5e00-0101 来封装报文。这样就能保证报文永远都是向主用设备转发了。是为虚实变换之间，合二为一也。

```
⊞ Frame 87: 60 bytes on wire (480 bits), 60 bytes captured (480 bits)
⊞ Ethernet II, Src: 00:00:00_06:19:01 (00:00:00:06:19:01), Dst: Broadcast (ff:ff:ff:ff:ff:ff)
⊟ Address Resolution Protocol (request/gratuitous ARP)
    Hardware type: Ethernet (0x0001)
    Protocol type: IP (0x0800)
    Hardware size: 6
    Protocol size: 4
    Opcode: request (0x0001)
    [Is gratuitous: True]
    Sender MAC address: IETF-VRRP-VRID_01 (00:00:5e:00:01:01)
    Sender IP address: 1.1.1.1 (1.1.1.1)
    Target MAC address: 00:00:00_00:00:00 (00:00:00:00:00:00)
    Target IP address: 1.1.1.1 (1.1.1.1)
```

至此，NAT Server 的三十二字真言阐释完毕。通过强叔的讲解，相信小伙伴们对 NAT Server 的正反 Server-map 表项作用，配置命令中的两个重要的参数 no-reverse、vrrp 的使用方法，以及多出口 NAT Server 的配置方法等都有了更加全面和深入的了解了吧。以后配置 NAT Server 时，小伙伴们应该能更加的得心应手了。

强叔提问

为什么 NAT Server 的公网地址和公网接口地址在同一个网段时，防火墙会发免费 ARP，而不在同一个网段时，防火墙不会发免费 ARP？

双剑合璧，无往不利——双向 NAT

经过前面几篇帖子的介绍，相信大家已经对源 NAT 和 NAT Server 有了相当了解。NAT 功能就像一个武林高手，可内可外，游刃有余，那么这“一内一外”能否配合使用呢？答案当然是肯定的！

如果需要**同时改变报文的源地址和目的地址**，就可以配置“源 NAT+NAT server”，华为防火墙称此类 NAT 配置为双向 NAT。这里要注意：双向 NAT 不是一个单独的功能，他仅仅是源 NAT 和 NAT Server 的组合。这里“组合”的含义是针对同一条流（例如外网主机访问内网服务器的流量），在其经过防火墙时同时转换报文的源地址和目的地址。大家千万不能理解为“防火墙上同时配置了源 NAT 和 NAT Server 就是双向 NAT”，这是不对的，因为源 NAT 和 NAT Server 可能是为不同流配置的。

之前介绍源 NAT 功能时，强叔为了更利于大家理解相关概念和原理，都是按照内网用户访问外网资源的思路进行组网设计和验证的。实际上，源 NAT 还可以根据报文的源地址和目的地址所在安全区域进行分类：

- 域间 NAT

报文的源地址和目的地址属于不同的安全区域。按照转换报文的方向，又可以分为以下两类：

5. NAT Inbound（外网访问内网）

报文由低安全级别的安全区域向高安全级别的安全区域方向传输时，基于源地址进行的转换。一般来说，NAT Inbound 都会和 NAT Server 配合使用。

6. NAT Outbound（内网访问外网）

报文由高安全级别的安全区域向低安全级别的安全区域方向传输时，基于源地址进行的转换。之前介绍的“内网用户访问外网资源”场景大多使用 NAT Outbound。

- 域内 NAT（内网访问内网）

报文的源地址和目的地址属于相同的安全区域。一般来说，域内 NAT 都会和 NAT Server 配合使用，单独配置域内 NAT 的情况较少见。

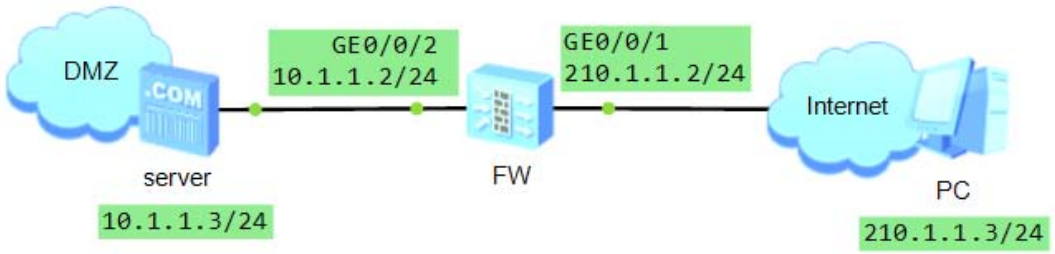
当域间 NAT 或域内 NAT 和 NAT Server 一起配合使用时，就实现了双向 NAT。当然，上述

内容的一个大前提就是：合理设置安全区域的级别并规划网络——内网设备属于 Trust 域（高级别），内网服务器属于 DMZ 域（中级别），外网设备属于 Untrust 域（低级别）。

双向 NAT 从技术和实现原理上讲并无特别之处，但是他和应用场景有着强相关性。究竟什么时候需要配置双向 NAT？配置后有什么好处？不配置双向 NAT 行不行？这都是实际规划和部署网络时需要思考的问题，且听强叔一一道来。

NAT Inbound+NAT Server

下图示意了一个最常见的场景：外网 PC 访问内网服务器，防火墙做服务器的网关。这个时候我们一般会用到的 NAT 技术是...（画外音：“强叔，我知道，是 NAT Server！这个场景不就是 NAT Server 的典型场景吗？”）没错，大家果然认真看了强叔之前的帖子！但是强叔下面要讲的是如何在这个场景中应用双向 NAT，以及这么做的好处，大家接着看吧。



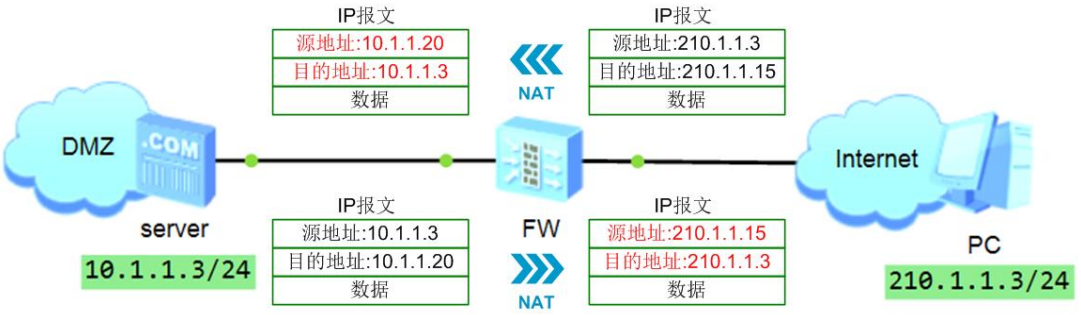
server 以公网 IP 对外提供服务，防火墙上配置 NAT Server，这个大家肯定没有疑问。同时，防火墙上配置 NAT Inbound，令 PC 以私网 IP 访问 server，这个大家可能有疑问，别着急，我们先来看看具体配置。

例 1 配置 NAT Inbound+NAT Server

```
#
 nat address-group 1 10.1.1.20 10.1.1.25   //地址池中的 IP 为私网 IP ，且和 server 的私网 IP 同网段
 nat server 0 global 210.1.1.15 inside 10.1.1.3
#
nat-policy interzone dmz untrust inbound
 policy 1
  action source-nat
  policy destination 10.1.1.3 0   //由于防火墙先做 NAT Server 转换，再做源 NAT 转换，所以此处的目的 IP 是 NAT Server 转换后的 IP
  address-group 1
```

这里 NAT Server 的配置和以前见过的类似，但是源 NAT 的配置和以前见过的不一样：以前地址池中配置的都是公网地址，而这次配置的却是私网地址。

我们通过下图再来看一下报文的地址转换过程：PC 访问 server 的流量经过防火墙时，目的地址（server 的公网地址）通过 NAT Server 转换为私网地址，源地址（PC 的公网地址）通过 NAT Inbound 也转换为私网地址，且和 server 的私网地址同网段，这样就同时转换了报文的源地址和目的地址，即完成了双向 NAT 转换。当 server 的回应报文经过防火墙时，再次做双向 NAT 转换，报文的源地址和目的地址均转换为公网地址。



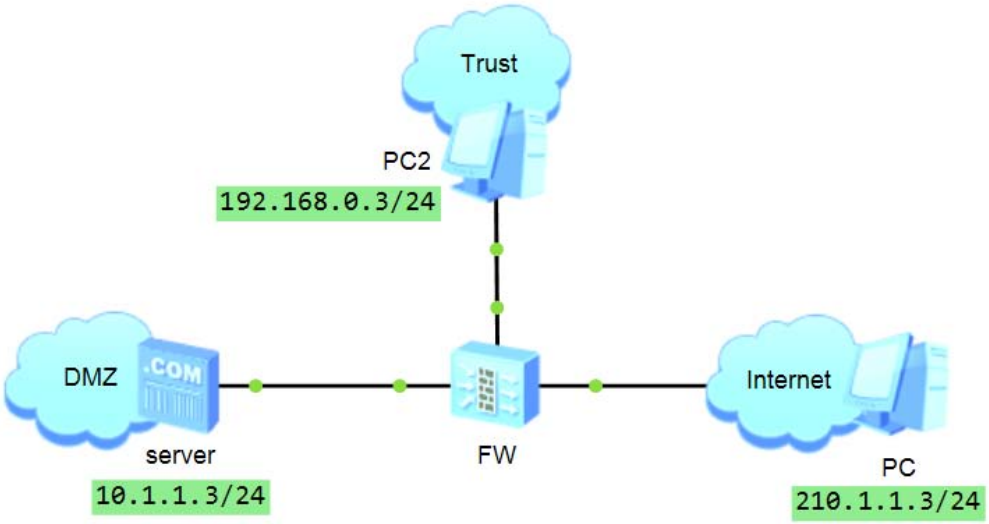
从 PC 上 ping server，通过防火墙上的会话表和 Server-map 表可以更清楚的看到双向 NAT 转换：PC 的地址通过 NAT Inbound 转换为私网地址，而 server 的地址也按照 NAT Server 的 Server-map 表转换为私网地址。

```
<SRG>display firewall session table

 Current Total Sessions : 5
  icmp  VPN:public --> public 210.1.1.3:56111[10.1.1.20:2063]-->210.1.1.15:2048[1
0.1.1.3:2048]
  icmp  VPN:public --> public 210.1.1.3:56367[10.1.1.20:2064]-->210.1.1.15:2048[1
0.1.1.3:2048]
  icmp  VPN:public --> public 210.1.1.3:56623[10.1.1.20:2065]-->210.1.1.15:2048[1
0.1.1.3:2048]
  icmp  VPN:public --> public 210.1.1.3:56879[10.1.1.20:2066]-->210.1.1.15:2048[1
0.1.1.3:2048]
  icmp  VPN:public --> public 210.1.1.3:57391[10.1.1.20:2067]-->210.1.1.15:2048[1
0.1.1.3:2048]
<SRG>display firewall server-map

 server-map item(s)
 --------------------------------------------------------------------------------
 Nat Server, any -> 210.1.1.15[10.1.1.3], Zone: ---
   Protocol: any(Appro: ---), Left-Time: --:--:--, Addr-Pool: ---
   VPN: public -> public

 Nat Server Reverse, 10.1.1.3[210.1.1.15] -> any, Zone: ---
   Protocol: any(Appro: ---), Left-Time: --:--:--, Addr-Pool: ---
   VPN: public -> public
```


好了，我们回过头来看为什么要配置 NAT Inbound 吧。如果不配置 NAT Inbound，行不行？行！不配置 NAT Inbound 并不影响 PC 访问 server。那配置 NAT Inbound 有什么好处？好处就是 server **上可以不用设置网关**，当然，前提条件是地址池中的地址需要和 server 的私网地址同网段。当 server 回应 PC 时，server 发现自己的地址和目的地址在同一网段，这时 server 就不会去查路由，而是发送 ARP 广播报文询问目的地址对应的 MAC 地址。由于目的地址是

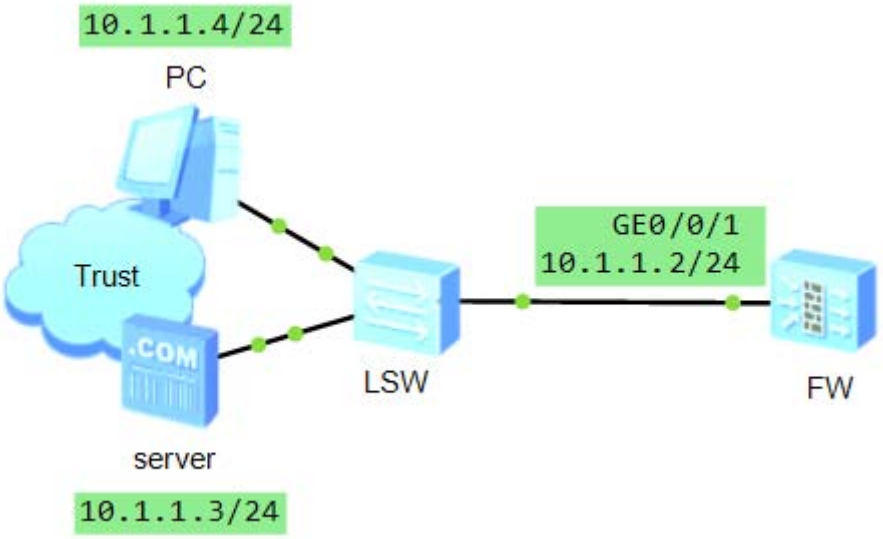
地址池中的地址，所以他没有对应的 MAC 地址，但是防火墙此时挺身而出，防火墙将自己与 server 直连接口的 MAC 地址发给 server，告诉 server“把回应报文给我吧”，所以回应报文将转发到防火墙上。由于 server 回应报文是通过二层转发，而不是三层转发，所以 server 上不用配置网关。也许有人说“配置网关还是挺方便的，不用配置 NAT Inbound 这么麻烦吧”如果只有一台服务器时，的确感受不到有什么好处，但是如果有几十台甚至上百台服务器需要配置或修改网关时，我们就会发现配置 NAT Inbound 是多么方便了。

如果对之前的组网做一点改变，增加一个 Trust 区域，该域内的 PC2 要访问 server 时，我们该如何配置双向 NAT 呢？和之前相比，报文的源地址所在安全域发生了变化，原来是 Untrust 域到 DMZ 域的报文（Inbound 方向），现在变成了 Trust 域到 DMZ 域的报文（Outbound 方向），所以双向 NAT 也变化为 NAT Outbound+NAT Server，它的转换原理和 NAT Inbound+NAT Server 完全一样，只不过源 NAT 的转换方向发生了改变而已。

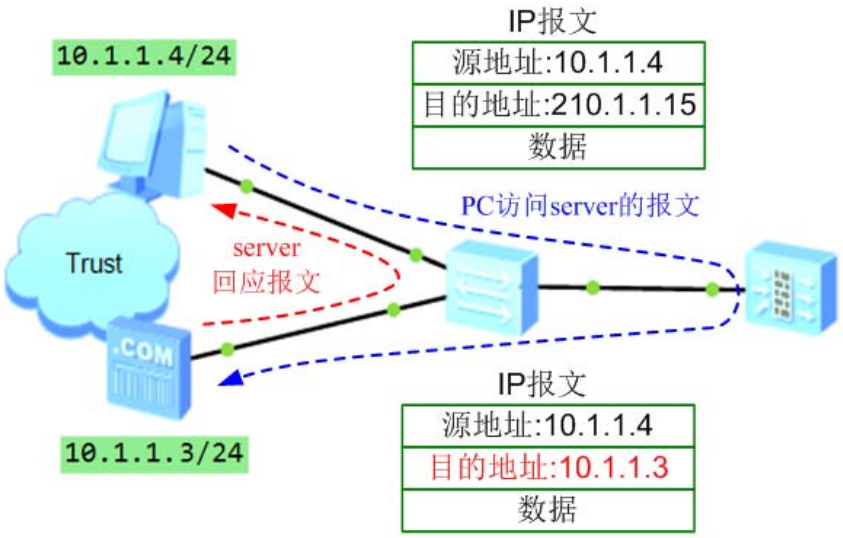


域内 NAT+NAT Server

域内 NAT 的场景多见于小型网络，如下图中的 PC 和 server 通过交换机与防火墙相连，管理员在规划网络时“偷懒”，将 PC 和 server 置于同一安全区域，并分配相同网段地址。



此时，如果希望 PC 像外网用户一样通过公网地址访问 server，就要在防火墙上配置 NAT Server。到此就配置完了吗？我们通过下图来看看吧：如果只配置了 NAT Server，报文到达防火墙后转换目的地址，server 回应报文时发现自己的地址和目的地址在同一网段，这就和之前分析的组网是同样情况了——server 通过二层转发报文，回应报文经交换机直接转发到 PC，不会经过防火墙转发！

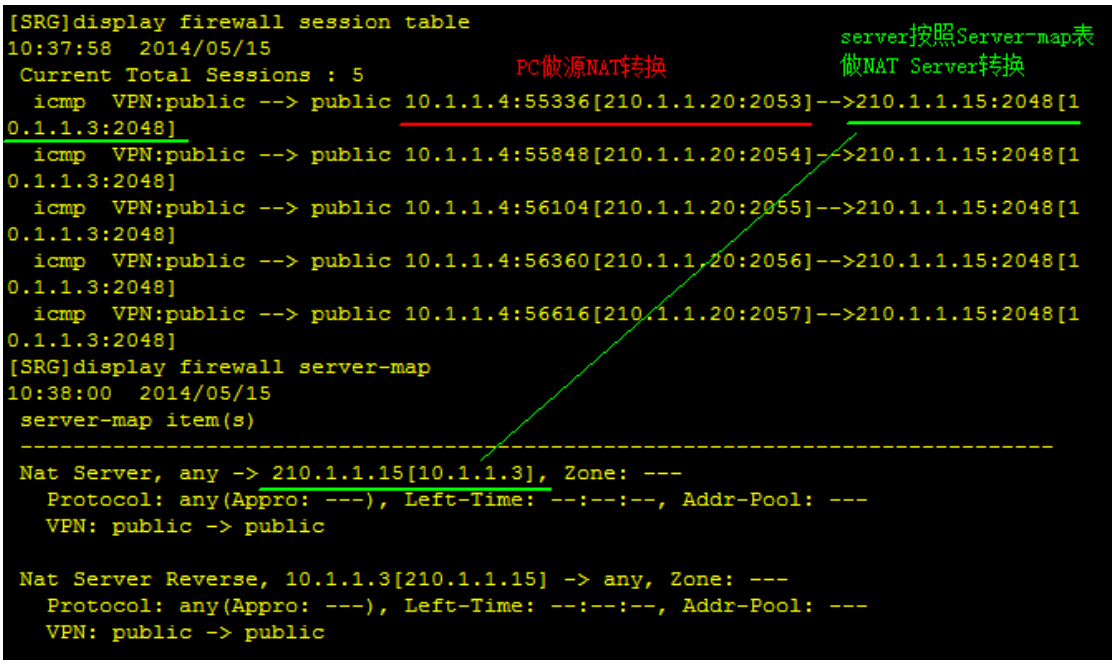


所以，如果希望提高内网的安全性，让回应报文也经过防火墙，就需要配置域内 NAT。下面列出了关键的配置步骤。地址池中的地址可以是公网地址，也可以是私网地址，关键是不能和 server 的私网地址在同一网段。域内 NAT 的配置和域间 NAT 几乎完全一样，只不过前者应用在域内做 NAT 转换，后者应用在域间做 NAT 转换。

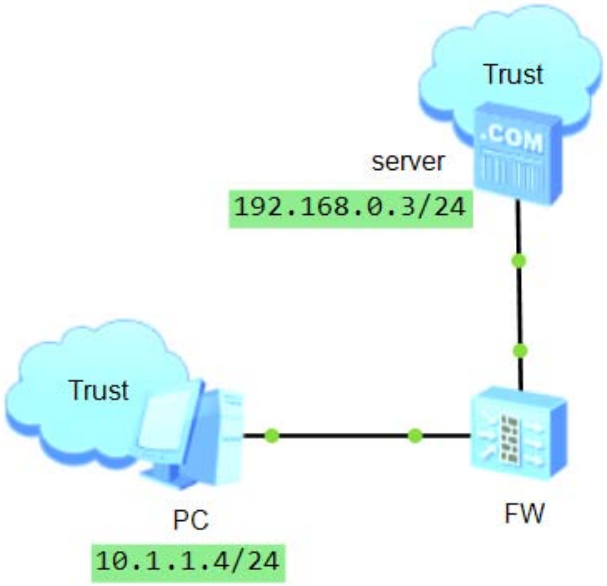
例 2 配置域内 NAT+NAT Server

```
#
 nat address-group 1 210.1.1.20 210.1.1.20
 nat server 0 global 210.1.1.15 inside 10.1.1.3
#
nat-policy zone trust   //注意是域内 NAT
 policy 1
  action source-nat
  policy destination 10.1.1.3 0   //此处的目的 IP 是 NAT Server 转换后的 IP
  address-group 1
```

从 PC 上 ping server，通过防火墙上的会话表和 Server-map 表可以看到：PC 的地址通过域内 NAT 转换为公网地址，server 的地址按照 NAT Server 的 Server-map 表转换为私网地址。双向 NAT 转换后，server 回复报文时发现自己的地址和目的地址不在同一网段，此时就需要查路由，通过三层转发报文，所以回应报文需经过防火墙转发。



如果在上面组网的基础上做一个变化，将 PC 和 server 分开，通过不同的接口和防火墙相连，此时应该如何配置双向 NAT 呢？在这个组网中所有报文都需要经过防火墙转发，只配置 NAT Server 是可以的。如果要配置双向 NAT，那么就是域内 NAT+NAT Server，具体配置方法和上面是类似的，此处就不再介绍了。



其实双向 NAT 的原理和配置并不复杂，关键是要想明白 NAT 转换的方向和转换后地址的作用，而不要纠结于转换后是公网地址还是私网地址。双向 NAT 并不是必配的功能，有时只配置源 NAT 或 NAT Server 就可以达到同样的效果，但是灵活应用双向 NAT 可以起到简化网络配置、方便网络管理的作用，也就达到了一加一大于二的效果！

强叔提问

对于域内 NAT，是否需要配置安全策略？如果不配置，且关闭缺省包过滤，PC 能否成功访问 server？

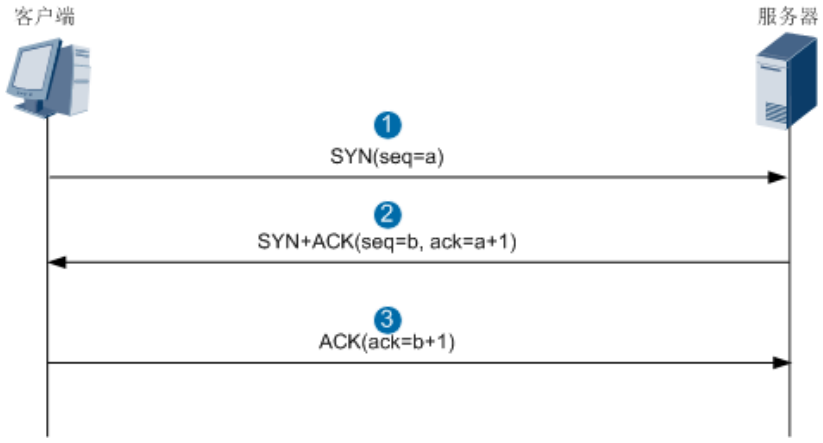
防火墙如何建立会话？

在《状态检测和会话机制》一篇中，我们学习了状态检测的工作原理，了解了会话中包含的五元组信息。贴子发出后，强叔收到了小伙伴们提出的各种问题：防火墙会为那些协议的报文建立会话呢？会话中就只包含五元组信息吗？状态检测功能在所有网络环境都适用吗？为了解答这些问题，强叔写了本篇贴子，作为状态检测和会话机制的番外篇，和大家进一步探讨状态检测和会话机制。

如今的数据通信网络已经是全 IP 时代，防火墙处理的也都是 IP 报文。根据 TCP/IP 协议族模型，在 IP 协议中，我们常用的协议有 TCP、UDP 和 ICMP 协议。那么下面我们就分别来看一下，对于 TCP、UDP 以及 ICMP 协议的报文，防火墙是如何建立会话的。

TCP

首先来看一下 TCP 协议。我们都知道，建立一个 TCP 连接，通信双方需要三次握手：



判断一个 TCP 连接的主要标志就是 SYN 报文，我们也把 SYN 报文称为 TCP 连接的首包。对于 TCP 协议，防火墙只有收到 SYN 报文，并且 SYN 报文通过了包括安全策略在内的各项安全机制的检查后，才会建立会话，后续的 TCP 报文匹配会话直接转发。如果防火墙没有收到 SYN 报文，只收到了 SYN+ACK 或 ACK 等后续报文，是不会创建会话的，并且会将这些报文丢弃。

下面我们就祭出 eNSP 模拟器，在防火墙上模拟一个典型的 TCP 协议的会话：HTTP 会话。网络拓扑如下：

PC 访问 Web 服务器，然后在防火墙上使用 **display firewall session table verbose** 命令可以看到会话正常建立，这里我们使用了 **verbose** 参数，通过这个参数可以看到会话的更多信息：

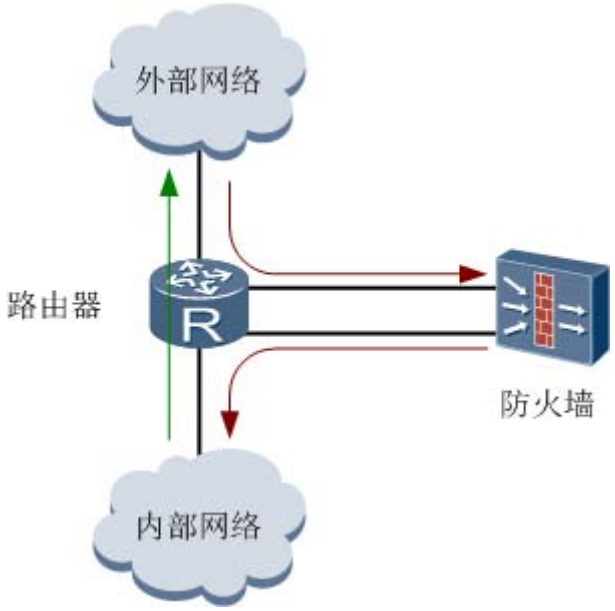
```
[SRG]display firewall session table verbose
09:35:01  2014/04/10
 Current Total Sessions : 1
  http  VPN:public --> public
  Zone: trust--> untrust  TTL: 00:00:10  Left: 00:00:05
  Interface: GigabitEthernet0/0/1  NextHop: 2.2.2.2  MAC: 54-89-98-b1-52-3f
  <--packets:4 bytes:465    -->packets:7 bytes:452
  1.1.1.1:2049-->2.2.2.2:80
```

除了我们上次介绍过的五元组信息之外，还有一些之前没见过的信息，我们简单介绍一下：

- Zone：表示报文在安全区域之间流动的方向，图中的信息表示报文是从 Trust 区域流向 Untrust 区域。
- TTL：表示该条会话的老化时间，这个时间到期后，这条会话也将会被清除。
- Left：表示该条会话剩余的生存时间。
- Interface：表示报文的出接口，报文从这个接口发出。
- NextHop：表示报文去往的下一跳的 IP 地址，本网络拓扑中是 Web 服务器的 IP 地址。
- MAC：表示报文去往的下一跳的 MAC 地址，本网络拓扑中是 Web 服务器的 MAC 地址。
- <--packets:4 bytes:465：表示会话反方向上的报文统计信息，即 Web 服务器向 PC 发送报文的个数和字节数。
- -->packets:7 bytes:452：表示会话正方向上的报文统计信息，即 PC 向 Web 服务器发送报文的个数和字节数。

在这里强叔要特意提一下，会话中“<--”和“-->”这两个方向上的报文统计信息非常重要，可以帮助我们定位网络故障。通常情况下，我们查看会话时发现只有“-->”方向有报文的统计信息，“<--”方向上的统计信息都是 0，那就说明 PC 发往 Web 服务器的报文顺利通过了防火墙，而 Web 服务器回应给 PC 的报文没有通过防火墙，双方的通信是不正常的。有可能是防火墙丢弃了 Web 服务器回应给 PC 的报文、或者是防火墙与 Web 服务器之间的网络出现故障、或者是 Web 服务器本身出现故障。这样我们就缩小了故障的范围，有利于快速定位故障。当然，凡事都有例外，在特殊的网络环境中，如果其中一个方向的报文统计信息是 0，双方的通信也有可能是正常的，这种特殊的网络环境是什么呢？这里先卖个关子，下面我们会讲到。

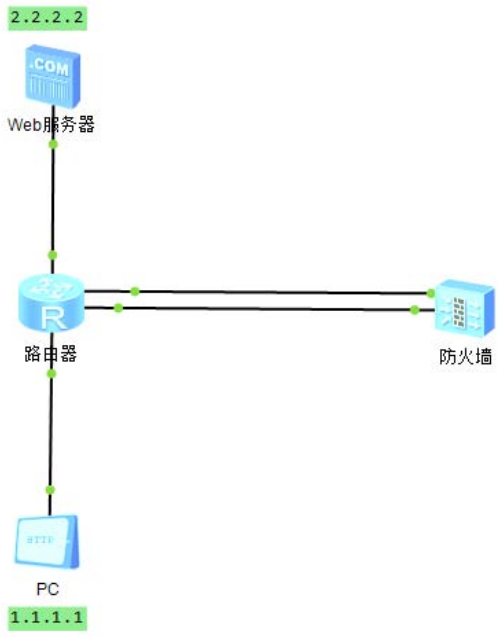
前面我们说过，如果防火墙没有收到 SYN 报文，只收到了 SYN+ACK 或 ACK 等后续报文，是不会创建会话的，并且会将这些报文丢弃。在正常情况下这样处理是没有问题的，但是在报文来回路经不一致的环境中，就会出现问题。



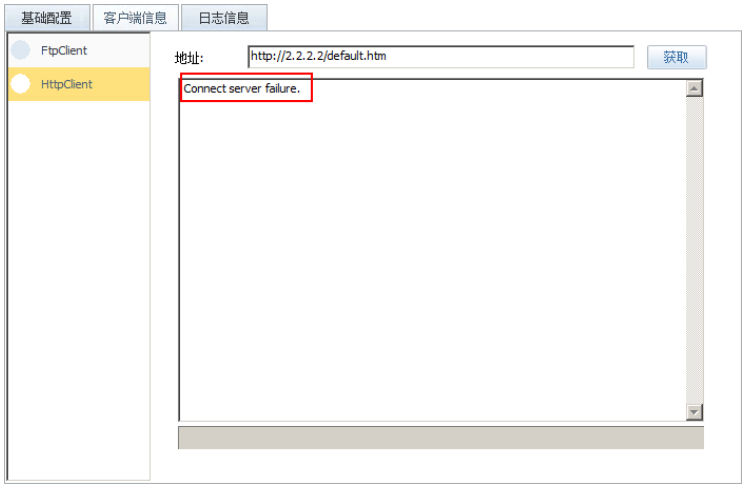
如上图所示，内部网络访问外部网络的报文直接通过路由器到达外部网络，而外部网络的回应报文，先经过路由器转发到防火墙，由防火墙处理后再转发到路由器，最后由路由器发送到内部网络。也就是防火墙无法收到 SYN 报文，只收到了 SYN+ACK 报文。这种通信双方交互的报文不同时经过防火墙的情况，叫做报文来回路经不一致。在这种网络环境中，防火墙收到 SYN+ACK 报文后，由于没有相应的会话，就会丢弃 SYN+ACK 报文，导致内部网络和外部网络之间的通信中断。

这种情况下该怎么办呢？别担心，防火墙早已经考虑到这个问题了，我们可以关闭防火墙的状态检测功能。关闭状态检测功能后，对于 TCP 协议，除了 SYN 报文之外，SYN+ACK、ACK 报文就都可以建立会话了，这样就不会导致通信中断。

下面我们使用 eNSP 来模拟一个报文来回路径不一致的网络环境，我们让 PC 访问 Web 服务器的报文通过路由器直接到达 Web 服务器，让 Web 服务器回应给 PC 的报文将会先转发到防火墙，然后再发送到 PC。网络拓扑如下：



我们先不关闭状态检测功能，让 PC 访问 Web 服务器，发现无法成功访问，在防火墙上也无法查看到会话信息：




```
<SRG>display firewall session table
11:55:00  2014/04/10
 Current Total Sessions : 0
```

此时在防火墙上使用 **display firewall statistic system discard** 命令查看丢包的情况，发现存在 **Session miss** 丢包：

```
<SRG>display firewall statistic system discard
11:27:07  2014/04/10
 Packets discarded statistic
                        Total packets discarded:          8
                  Session miss packets discarded:         8
```

这表示防火墙因为无法找到会话而将报文丢弃。因为防火墙只收到了服务器回应的

SYN+ACK 报文，没有收到 SYN 报文，也就没有相应的会话，所以 SYN+ACK 报文被丢弃。接下来我们使用 **undo firewall session link-state check** 命令关闭状态检测功能，然后再让 PC 访问 Web 服务器，发现可能访问成功，在防火墙上也可以查看到会话信息：

```
[SRG]display firewall session table verbose
11:57:08  2014/04/10
 Current Total Sessions : 1
  tcp  VPN:public --> public
  Zone: untrust--> trust  TTL: 00:00:10  Left: 00:00:03
  Interface: GigabitEthernet0/0/0  NextHop: 3.3.3.1  MAC: 54-89-98-15-2c-1f
  <--packets:0 bytes:0  -->packets:5 bytes:509
  2.2.2.2:80-->1.1.1.1:2052
```

在会话信息中，“<--”方向的统计信息是 0，只有“-->”方向存在统计信息，这就说明只有服务器回应的 SYN+ACK 报文经过了防火墙。由此我们得出结论，关闭状态检测功能后，防火墙收到 SYN+ACK 报文后也会建立会话，PC 和 Web 服务器之间的通信不会中断。

在报文来回路径不一致的网络环境中，我们在防火墙上关闭状态检测功能后，会话中的一个方向上的报文统计信息是 0，此时双方的通信也是正常的，这就是我们上面所说的特殊的网络环境。可见在实际的网络环境中，我们还是要具体情况具体分析。

UDP

接下来我们看一下 UDP 协议。UDP 协议不同于 TCP 协议，它是没有连接状态的协议。对于 UDP 协议，防火墙收到 UDP 报文后，无论状态检测功能是开启还是关闭状态，只要 UDP 报文通过了包括安全策略在内的各项安全机制的检查，防火墙都会建立会话。

ICMP

然后是 ICMP 协议。一提到 ICMP 协议，我们首先就会想到 Ping。Ping 用来测试网络中的另一台设备是否可达，是我们在日常维护中经常会用到的操作。执行 Ping 操作的一方会发送 Ping 回显请求报文（Echo request），收到该请求报文后，响应一方会发送 Ping 回显应答报文（Echo reply）。

对于 Ping 报文，在开启状态检测功能时，防火墙只有收到 Ping 回显请求报文，并且 Ping 回显请求报文通过了包括安全策略在内的各项安全机制的检查后，才会建立会话；如果防火墙没有收到 Ping 回显请求报文，只收到了 Ping 回显应答报文，是不会创建会话的，并且会将 Ping 回显应答报文丢弃。在关闭状态检测功能时，防火墙收到 Ping 回显请求报文和 Ping 回显应答报文，都会创建会话。下面是在报文来回路径不一致的网络环境中，防火墙上关闭了状态检测功能后，Ping 回显应答报文生成的会话信息：

```
[SRG]display firewall session table verbose
14:44:09  2014/04/10
 Current Total Sessions : 5
  icmp  VPN:public --> public
  Zone: untrust--> trust  TTL: 00:00:20  Left: 00:00:11
  Interface: GigabitEthernet0/0/0  NextHop: 3.3.3.1  MAC: 54-89-98-15-2c-1f
  <--packets:0 bytes:0  -->packets:1 bytes:60
  2.2.2.2:2048-->1.1.1.1:45117
```

而对于其他类型的 ICMP 报文，无论状态检测功能是开启还是关闭状态，只要这些报文通过了包括安全策略在内的各项安全机制的检查，防火墙都会转发报文，不建立会话。

最后我们再来总结一下防火墙对 TCP、UDP 和 ICMP 协议的报文创建会话的情况，如下表所示。当然，前提还是这些报文要通过防火墙上包括安全策略在内的各项安全机制的检查，然后才会创建会话。

协议		开启状态检测功能	关闭状态检测功能
TCP	SYN 报文	创建会话，转发报文	创建会话，转发报文
	SYN+ACK、ACK 报文	不创建会话，丢弃报文	创建会话，转发报文
UDP		创建会话，转发报文	创建会话，转发报文
ICMP	Ping 回显请求报文	创建会话，转发报文	创建会话，转发报文
	Ping 回显应答报文	不创建会话，丢弃报文	创建会话，转发报文
	其他 ICMP 报文	不创建会话，转发报文	不创建会话，转发报文

另外，还有一些特殊的报文，防火墙转发这些报文时不会创建会话，这些特殊的报文包括：

- RAW IP（即 IP Protocol 字段没有值）报文不创建会话
- OSPF、RIP 和 ISIS 路由协议报文不创建会话
- IGMP（组播）报文不创建会话
- 二层模式下目的 MAC 为 Unkown MAC 而需要在 VLAN 内广播的报文不创建会话

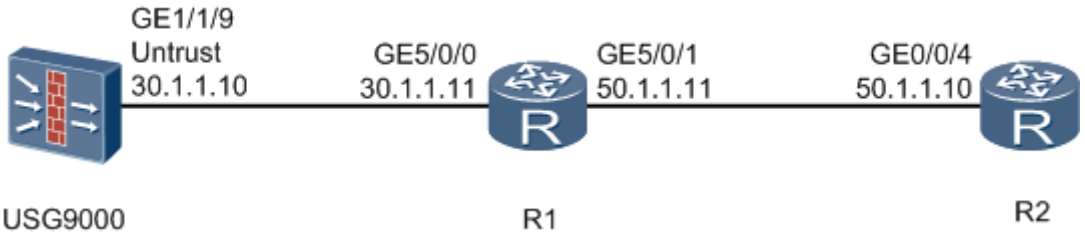
通过上面的介绍，我们知道了防火墙在开启或者关闭状态检测功能的情况下，对 TCP、UDP 和 ICMP 协议报文的不同处理方式。相信大家对状态检测和会话机制有了进一步的了解，希望大家继续关注强叔侃墙系列连载贴。

当安全策略遇上 OSPF

在《安全策略初体验》一篇中，强叔提到，路由协议一般是不受安全策略控制的，防火墙直接允许路由协议的报文通过。强叔还提到，这和具体产品实现有关，不同型号的产品会有差异。大家看完帖子后感觉有点语焉不详，纷纷向强叔咨询这个问题。今天，强叔就使用华为防火墙 USG9000（软件版本为 V300R001）来实际验证一把，看一看当安全策略遇上 OSPF 路由协议时，会发生什么事情。

需要说明一点，本篇验证的是防火墙本身参与到 OSPF 路由计算的场景，即验证防火墙接口所在安全区域与 Local 区域之间是否需要开启安全策略。而防火墙本身不参与 OSPF 路由计算，只是二层转发 OSPF 路由报文的场景中，如果防火墙的接口属于不同的安全区域，则需要开启安全区域之间的安全策略，允许 OSPF 路由报文通过。

我们使用一台 USG9000 防火墙和两台路由器搭建一个简单的网络环境，网络拓扑如下图所示：



根据下表中的数据，在 USG9000 上配置接口的 IP 地址、将接口加入安全区域、开启 OSPF 功能，以及在路由器 R1 和 R2 上配置接口的 IP 地址、开启 OSPF 功能。具体配置过程在这里不再赘述。

配置项	USG9000	R1	R2
接口	# interface GigabitEthernet1/1/9 ip address 30.1.1.10 255.255.255.0 #	# interface GigabitEthernet5/0/0 ip address 30.1.1.11 255.255.255.0 # interface GigabitEthernet5/0/1 ip address 50.1.1.11 255.255.255.0 #	# interface GigabitEthernet0/0/4 ip address 50.1.1.10 255.255.255.0 #

安全区域	# firewall zone untrust set priority 5 add interface GigabitEthernet1/1/9 #	-	-
OSPF	# ospf 1 area 0.0.0.1 network 30.1.1.0 0.0.0.255 #	# ospf 1 area 0.0.0.1 network 30.1.1.0 0.0.0.255 network 50.1.1.0 0.0.0.255 #	# ospf 1 area 0.0.0.1 network 50.1.1.0 0.0.0.255 #

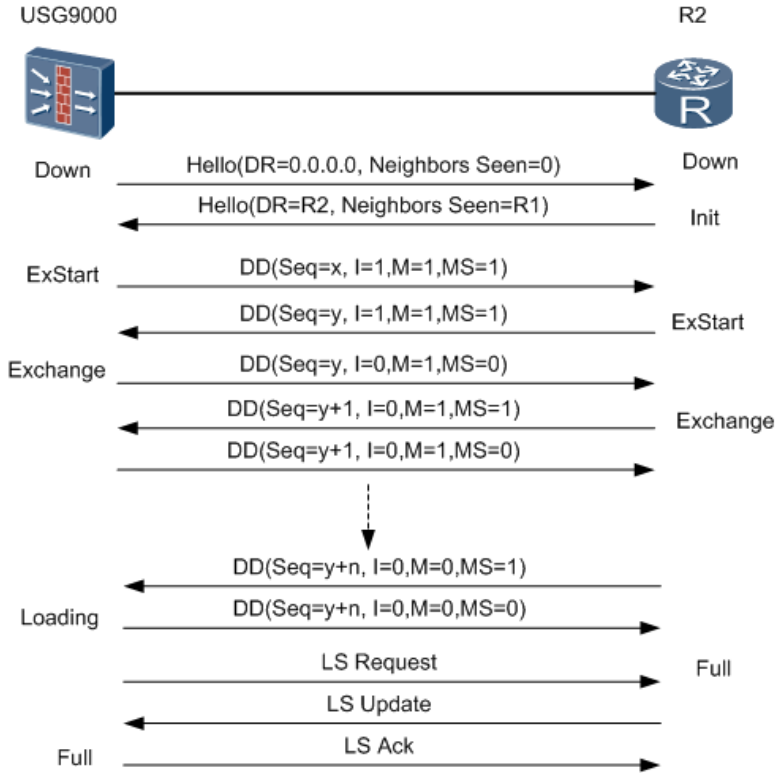
默认情况下，防火墙上没有开启 GE1/1/9 接口所在的 Untrust 区域和 Local 区域之间的安全策略，这两个区域之间不允许报文通过。我们在防火墙上使用 **display ospf peer** 命令查看 OSPF 的邻接关系：

```
[USG9000]display ospf peer
11:29:40  2014/04/17

         OSPF Process 1 with Router ID 88.88.8.8
                 Neighbors

 Area 0.0.0.1 interface 30.1.1.10(GigabitEthernet1/1/9)'s neighbors
 Router ID: 128.18.140.6     Address: 30.1.1.11
   State: ExStart  Mode:Nbr is  Slave  Priority: 1
   DR: 30.1.1.11  BDR: 30.1.1.10  MTU: 0
   Dead timer due in 27  sec
   Retrans timer interval: 0
   Neighbor is up for 00:00:00
   Authentication Sequence: [ 0 ]
```

在路由器 R1 上使用 **display ospf peer** 命令查看 OSPF 的邻接关系：

```
[R1]display ospf peer
12:32:09  2014/04/17

        OSPF Process 1 with Router ID 128.18.140.6
                Neighbors

 Area 0.0.0.1 interface 30.1.1.11(GigabitEthernet5/0/0)'s neighbors
 Router ID: 88.88.8.8        Address: 30.1.1.10        GR State: Normal
   State: ExStart  Mode:Nbr is  Slave  Priority: 1
   DR: 30.1.1.11  BDR: 30.1.1.10  MTU: 0
   Dead timer due in 30  sec
   Neighbor is up for 00:00:00
   Authentication Sequence: [ 0 ]

                Neighbors

 Area 0.0.0.1 interface 50.1.1.11(GigabitEthernet5/0/1)'s neighbors
 Router ID: 128.18.140.2     Address: 50.1.1.10        GR State: Normal
   State: Full  Mode:Nbr is  Slave  Priority: 1
   DR: 50.1.1.10  BDR: 50.1.1.11  MTU: 0
   Dead timer due in 34  sec
   Neighbor is up for 00:09:28
   Authentication Sequence: [ 0 ]
```

在 USG9000 和 R1 上看到的 OSPF 邻接状态都为 ExStart，我们根据下面的 OSPF 邻接关系建立过程示意图，发现 OSPF 邻接关系没建立起来，因为 USG9000 和 R1 之间没有成功交换 DD（Database Description）报文。



此时，我们怀疑有可能是 USG9000 丢弃了 DD 报文。在 USG9000 上使用 **display firewall statistic system discarded** 命令查看丢包信息：

```
[USG9000]display firewall statistic system discarded
15:46:19  2014/04/17
Packets discarded statistic on slot 2 CPU 3
                                  Total packets discarded : 69

                            Interzone miss packets discarded : 38

                                 Total deny bytes discarded : 1,612
                             Default deny packets discarded : 31
```

上面的信息表示缺省包过滤将报文丢弃，而且被丢弃报文的个数还在不断增长，说明 OSPF 模块在不断地尝试发送 DD 报文，但都被安全策略模块丢弃了。

接下来我们在 USG9000 上开启 Local 区域和 Untrust 区域之间的安全策略，允许 OSPF 报文通过。注意，因为 USG9000 既要发送 DD 报文又要接收 DD 报文，所以 Inbound 和 Outbound 方向上的安全策略都要开启。如下：

说明

*这里为了精确匹配 OSPF 协议，我们使用了系统提供的 **ospf** 服务集，如果防火墙中没有提供这个服务集，我们可以自己创建一个服务集，协议号设置为 89 即可。*

```
#
policy interzone local untrust inbound
 policy 1
  action permit
  policy service service-set ospf
#
policy interzone local untrust outbound
 policy 1
  action permit
  policy service service-set ospf
#
```

然后分别在 USG9000 和 R1 上使用 **display ospf peer** 命令查看 OSPF 的邻接关系（可能需要等待几分钟的时间才会出现下面的结果，或者可以使用 **reset ospf process** 重启 OSPF 进程，就可以很快看到结果）：

```
[USG9000]display ospf peer
16:48:38  2014/04/17

        OSPF Process 1 with Router ID 88.88.8.8
                Neighbors

 Area 0.0.0.1 interface 30.1.1.10(GigabitEthernet1/1/9)'s neighbors
 Router ID: 128.18.140.6     Address: 30.1.1.11
   State: Full  Mode:Nbr is  Master  Priority: 1
   DR: 30.1.1.11  BDR: 30.1.1.10  MTU: 0
   Dead timer due in 36  sec
   Retrans timer interval: 4
   Neighbor is up for 00:00:51
   Authentication Sequence: [ 0 ]
```

```
[R1]display ospf peer
17:48:02  2014/04/17

        OSPF Process 1 with Router ID 128.18.140.6
                Neighbors

 Area 0.0.0.1 interface 30.1.1.11(GigabitEthernet5/0/0)'s neighbors
 Router ID: 88.88.8.8        Address: 30.1.1.10        GR State: Normal
   State: Full  Mode:Nbr is  Slave  Priority: 1
   DR: 30.1.1.11  BDR: 30.1.1.10  MTU: 0
   Dead timer due in 35  sec
   Neighbor is up for 00:01:30
   Authentication Sequence: [ 0 ]

                Neighbors

 Area 0.0.0.1 interface 50.1.1.11(GigabitEthernet5/0/1)'s neighbors
 Router ID: 128.18.140.2     Address: 50.1.1.10        GR State: Normal
   State: Full  Mode:Nbr is  Slave  Priority: 1
   DR: 50.1.1.11  BDR: 50.1.1.10  MTU: 0
   Dead timer due in 30  sec
   Neighbor is up for 01:35:43
   Authentication Sequence: [ 0 ]
```

此时可以看到 OSPF 邻接建立成功，同时 USG9000 上已经存在了通过 OSPF 路由协议学习到的去往 50.1.1.0 这个网段的路由：

```
[USG9000]display ip routing-table protocol ospf
16:55:17  2014/04/17
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Public routing table : OSPF
        Destinations : 2          Routes : 2

OSPF routing table status : <Active>
        Destinations : 1          Routes : 1

Destination/Mask    Proto  Pre  Cost       Flags NextHop         Interface

      50.1.1.0/24  OSPF   10   2            D   30.1.1.11       GigabitEthernet
1/1/9
```

我们来总结一下，对于防火墙 USG9000 来说，需要开启接口所在安全区域和 Local 区域之间的安全策略，允许 OSPF 报文通过，这样防火墙才能和相连的设备正常建立邻接关系。

实际上，我们可以从另外一个角度来考虑这个问题：单播报文和组播报文。对于防火墙来说，

一般情况下，单播报文是要经过安全策略的检查，所以需要开启安全策略允许报文通过；而组播报文不经过安全策略的检查，也就不需要开启相应的安全策略。

那么在 OSPF 中，哪些报文是单播哪些报文是组播呢？不同的网络类型，OSPF 报文的发送形式也不相同，如下表所示。

说明

我们可以在接口上执行 ***ospf network-type*** *命令来修改 OSPF 的网络类型。*

网络类型	Hello	Database Description	Link State Request	Link State Update	Link State Ack
Broadcast	组播	单播	单播	组播	组播
P2P	组播	组播	组播	组播	组播
NBMA	单播	单播	单播	单播	单播
P2MP	组播	单播	单播	单播	单播

从表中可以看出，网络类型是 Broadcast 类型时，OSPF 报文中的 DD 报文和 LSR 报文是单播报文，需要开启安全策略；网络类型是 P2P 时，OSPF 报文都是组播报文，因此无需开启安全策略。NBMA 和 P2MP 类型也是同理。

最后再次提醒，上面的验证结论基于华为防火墙 USG9000 V3R1 版本，并不适用于其他型号的防火墙产品。本篇主要还是提供一种验证思路，大家可以参考强叔的操作过程来验证华为 USG2000、USG5000 和 USG6000 系列防火墙产品，看一看在这些防火墙产品上是否需要开启安全策略。另外，在实际网络环境中，如果防火墙上运行的 OSPF 状态不正常，大家也可以从安全策略这个角度入手，检查是不是由于没有开启安全策略而导致的。

揭密华为防火墙 NAT 地址复用专利技术

提到多对多、多对一的 NAT（多个私网地址转换为多个或一个公网地址），就不能回避公网地址利用率的问题。“华为防火墙一个公网 IP 突破了 65535 端口限制，理论上能够无限制进行 NAT 转换”这个结论在江湖上已经广为流传，接触过华为防火墙的兄弟可能早有听闻。这正是华为防火墙十年前申请的一项专利技术的应用。

至此处，原谅强叔回忆下此专利发明人的大牛风采。几前路遇隐居的大牛下山，强叔敬仰地双手抱拳：“哥你已不在江湖多年，江湖中仍然流传着哥的传说。”——强叔还被称为小强的时候，大牛已经名满防火墙。此时大牛仍然腼腆一笑，道声旧日小事飘然去也，唯强叔口水流了一地。之后，强叔又马不停蹄至襄阳，请教如今的 NAT 传人后，才算对此专利解读稍有眉目。围观兄弟且看，强叔接下来揭密华为防火墙 NAT 地址复用技术。

在《源 NAT》一节，强叔提到：“防火墙在应用源 NAT 功能时就是从地址池中挑选出一个公网 IP，然后对私网 IP 进行转换。**挑选哪个公网 IP 是随机的，和配置时的顺序、IP 大小等因素都没有关系。**”这其实就是我们看到的外部招式，所谓内功心法不外露的。

那么在进行源 NAT 时，究竟是怎样从地址池中挑选出**公网 IP 地址资源**进行分配的呢？

请允许强叔引入Hash 算法的概念，这应该是大家常听说的一种广泛应用于程序编写的方法。一句话解释，就是把一种任意长度的信息进行压缩映射，成为某一固定长度的信息。例如，把 3000 个私网 IP 映射成 100 个公网 IP，也就是从地址池挑选出公网 IP 地址资源进行分配的过程，资料中也偶尔会提到的“基于源地址 Hash”。

Hash 的具体规则或者说算法是什么呢？很灵活，自行设定，但是根据要达成的目标和计算结果，可以优选。在这里，我们使用的算法也比较简单—取模运算。

估计很多不写代码兄弟的数学都已经还给老师了，但是提起求余运算应该都知道，思路如下：

> X=地址池中的地址个数
> Y=内网用户的 IP 地址（转换为 32 位二进制数值）
> 将 X、Y 进制统一，使用 Y/X 的余数对应分配公网 IP 地址资源。
> 余数为 0，则选择地址池连续地址的第一个；余数为 1，则选择第二个……余数不可能大于地址池个数，最大的余数刚好对应池址池的最后一个地址。

示例如下：

内网用户 10.1.1.1-10.10.10.1，NAT 地址池 202.169.1.1-202.169.1.5

X=5

Y=10.1.1.1

10.1.1.1----->00001010 00000001 00000001 00000001----->167837953

不用计算可目测余数为 3，选择地址池的第四个资源即 202.169.1.4

至此，内网用户的源 IP 地址已经可以全部被分配了对应的地址池中的公网 IP 地址资源，并且保证了每个内网的 IP 地址在转换时每次访问外部网络时始终转换为同一个公网 IP 地址。按照上述运算后，一部分内网用户就会被分配相同的公网 IP 地址资源。下一步，我们来研究下如何分配端口资源。

其实强叔对于将复杂一点的数学运算表达清楚没有太多信心，决定到最后不得已的时候再引出来，下面端口分配先从防火墙的实现原理讲起，定性区分出端口分配带来的影响。

在《源 NAT》一节中，我们已介绍了 NAPT 情况建立的会话表。

例如，内网 10.1.1.1 和 10.1.1.2 用户分别 ping 220.180.20.50 服务器，在 USG 上通过公网地址 202.169.1.1 进行 NAT 转换，查询会话表显示如下：

```
[USG9000] display firewall session table
 Current total sessions: 2
 Slot: 5 CPU: 1
 icmp VPN: public --> public 10.1.1.1:1280[202.169.1.1:10298] --> 220.180.20.50:2048
 icmp VPN: public --> public 10.1.1.2:1280[202.169.1.1:6103] --> 220.180.20.50:2048
```

由于内网不同用户的 IP 地址和端口必不相同，仅使用“源地址+源端口”二元组信息在 USG 上即可标识一条数据流，来建立正向 NAT 地址转换。**而进行反向地址还原时，使用“源地址+源端口+目的地址+目的端口+协议”五元组信息唯一标识一条数据流。**

那么，根据会话表实现机制，**只要内网不同用户访问“目的地址+目的端口+协议”三元组中的任一参数不同时，即使将地址池中同一公网地址的同一端口同时分配给内网多个用户时，也不会产生冲突。**

示例如下：

编号	内网源地址	内网源端口	NAT 地址	NAT 端口	目的地址	目地端口	协议
1	10.1.1.1	80	202.169.1.1	8080	www.baidu.com	80	http
2	10.1.1.2	80	202.169.1.1	8080	www.sohu.com	80	http

因此，只要内网不同用户访问“目的地址+目的端口+协议”三元组中的任一参数不同时，唯一的 NAT 地址和端口可以反复利用，不受 65535 端口的限制。此时无论端口如何分配，都不会产生问题。

围观兄弟接下来肯定就有问题等着强叔了：在地址池只有**一个公网 IP 情况下**，如果**访问相同的“目的地址+目的端口+协议”三元组**时怎么办？现实可能就是这么残酷……

在内网不同用户访问相同的“目的地址+目的端口+协议”三元组时，不能被分配给相同的 NAT 地址和端口资源，这是因为一旦完全相同，访问的目的主机会发现出现同样的“源地址+源端口”访问本主机的同一“目的地址+端口+协议”，目的主机无法正确回应甚至可能会判定为受到攻击。

关键点来了！因此，保证内网不同用户、访问相同的“目的地址+目的端口+协议”三元组时，不能被分配到相同地址的相同端口资源是关键。这就要引入冲突检测机制。

其实在分配端口的法则上，我们仍然使用 Hash 算法，使得在内存占用合理情况下、尽可能保证被分配到相同公网地址的用户、其被分配的端口尽量不一致，简化的基本思路如下：

> Z=地址池地址 ⊕ 访问的目的地址 ⊕ 目的端口 ⊕ 协议(按一定规则和对应关系异或运算)
>
> 得到分配的端口后，根据会话表判断是否该端口已经被分配。如果检测已经被分配，则执行 X++运算，重新分配端口。
>
> 例如，计算出 3000 端口，但检测到 3000 端口已经被使用，那么在其之上加 1，分配 3001 端口。如果 3001 也被占用，继续执行运算，直至找到未使用端口。

这就保证了，不同内网户访问相同的 “目的地址+目的端口+协议”三元组，不会被分配到相同的 NAT 地址的端口资源。而且会话表是会实时老化的，被分配过的端口在会话表老化后会重新被利用，因此，概率上端口也不会受到限制。

除非最极端的情况发生：超过 64k 的内网用户，**同一时刻**、向外网**同一目的主机**的**同一端口**、**采用同样协议**发起链接。不过兄弟们觉不觉得很面熟，这看起来就是发起了传统而典型的 DDoS 攻击了吧-_-

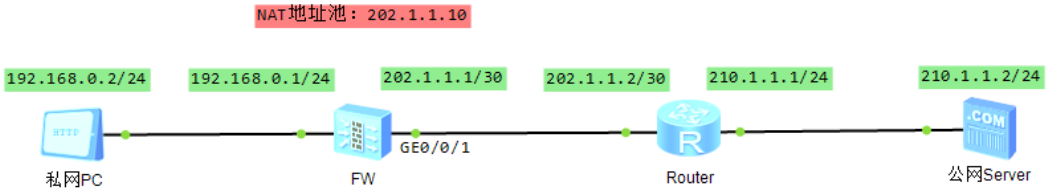
好了，揭密到此，强叔已经全盘兜出了，大家是否已经了然于胸了？

其实原专利本身是晦涩难懂的，华为防火墙也在这么多年的发展中进行着不断演进，强叔只是对其基本实现原理进行了简单大意解读。若有对原专利感兴趣的小伙伴可自行谷歌：CN1567907A 一种网络地址资源的利用方法。

配置 NAT 时为什么要同时配置黑洞路由？

在 NAT 篇和拍案惊奇系列中，强叔多次提到配置 NAT 的同时要配置黑洞路由，避免路由环路，很多人对此不太理解，今天强叔就来为大家详细介绍其中缘由。

首先我们用 eNSP 模拟一个典型的源 NAT 环境：

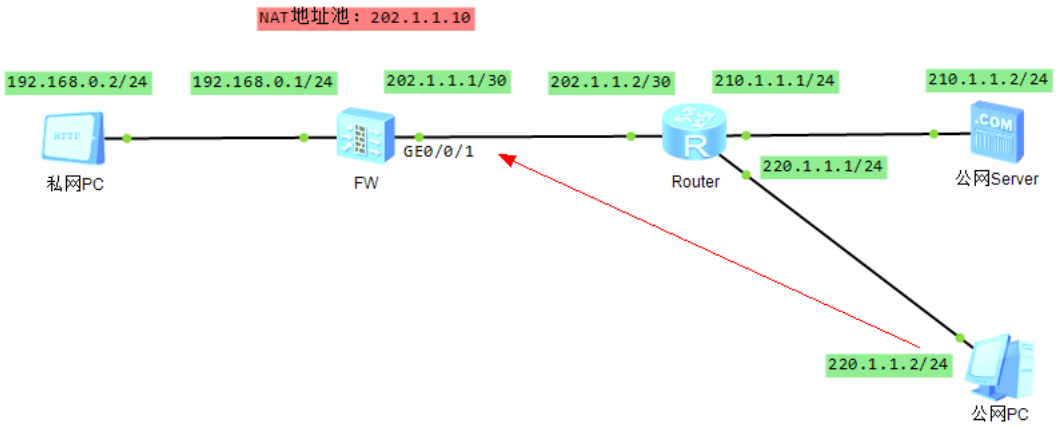


NAT 地址池地址是 202.1.1.10，防火墙与路由器互联接口地址的掩码是 30 位，与 NAT 地址池地址**不在同一网段**。防火墙上配置了一条缺省路由，下一跳是 202.1.1.2，这样就能把私网 PC 访问公网 Server 的报文送到路由器。为了保证公网 Server 的回程报文能够顺利到达防火墙，路由器上还要配置了一条到 NAT 地址池地址的路由。另外，防火墙上的 NAT 策略和安全策略也都配置完成了，在此不再赘述。

正常情况下，私网 PC 访问公网上的服务器 Server，生成会话表，源地址也进行了转换，一切都没有问题。

```
<FW>display firewall session table
09:03:54  2014/05/13
 Current Total Sessions : 1
  http  VPN:public --> public 192.168.0.2:2056[202.1.1.10:2055]-->210.1.1.2:80
```

此时，如果公网上的一台 PC，主动访问防火墙上的 NAT 地址池地址，会发生什么情况呢？



我们在公网 PC 上执行 **ping 202.1.1.10** 命令，发现不能 ping 通：

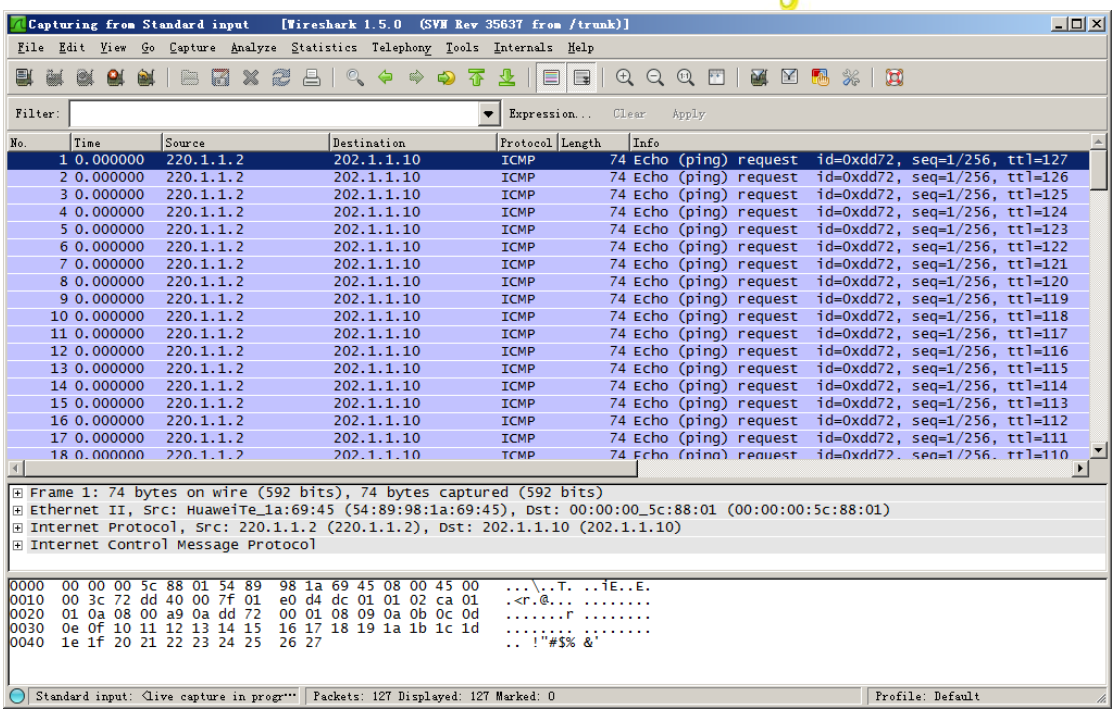
```
PC>ping 202.1.1.10

Ping 202.1.1.10: 32 data bytes, Press Ctrl_C to break
Request timeout!
Request timeout!
Request timeout!
Request timeout!
Request timeout!

--- 202.1.1.10 ping statistics ---
  5 packet(s) transmitted
  0 packet(s) received
  100.00% packet loss
```

显然，这是正常的结果。因为 NAT 地址池只有在转换私网地址的时候才会用到，也就是说，私网 PC 必须先发起访问请求，防火墙收到该请求后才会为其转换地址，NAT 地址池地址并不对外提供任何单独的服务。所以当公网 PC 主动访问 NAT 地址池地址时，报文到达防火墙后，无法匹配会话表，防火墙肯定就会把报文丢弃了。

但实际情况远没有这么简单，我们在防火墙的 GE0/0/1 接口抓包，然后再次在公网 PC 上执行 **ping 202.1.1.10** 命令，这次我们使用**-c** 参数，只发送一个 ping 报文：

```
PC>ping 202.1.1.10 -c 1

Ping 202.1.1.10: 32 data bytes, Press Ctrl_C to break
Request timeout!

--- 202.1.1.10 ping statistics ---
  1 packet(s) transmitted
  0 packet(s) received
  100.00% packet loss
```

GE0/0/1 接口上的抓包信息如下：



嚯！不看不知道，一看吓一跳，居然抓到了这么多 ICMP 报文。经过分析发现，报文的 TTL 值逐一递减，直到变为 1。我们都知道，TTL 是报文的生存时间，每经过一台设备的转发，TTL 的值减 1，当 TTL 的值为 0 时，就会被设备丢弃。这说明公网 PC 主动访问 NAT 地址池地址的报文，在防火墙和路由器之间相互转发，直到 TTL 变成 0 之后，被最后收到该报文的那台设备丢弃。

我们来梳理一下整个过程：

1. 路由器收到公网 PC 访问 NAT 地址池地址的报文后，发现目的地址不是自己的直连网段，因此查找路由，发送到防火墙。

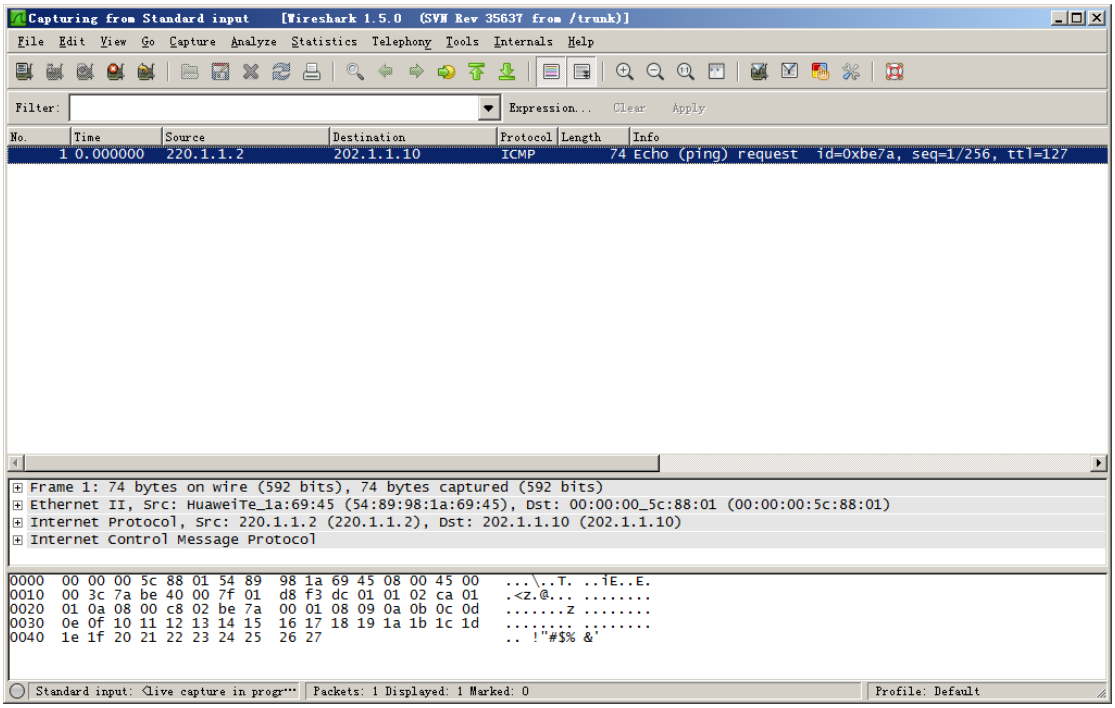
2. 防火墙收到报文后，该报文不属于私网访问公网的回程报文，无法匹配到会话表，同时目的地址也不是自己的直连网段（防火墙没有意识到该报文的目的地址是自己的 NAT 地址池地址），只能根据缺省路由来转发。因为报文从同一接口入和出，相当于在同一个安全区域流动，缺省情况下也不受安全策略的控制，就这样报文又从 GE0/0/1 接口送出去了。

3. 路由器收到报文后，查找路由，还是发送至防火墙，如此反复。这个可怜的报文像皮球一样被两台设备踢来踢去，最终被残忍丢弃，憾别网络… …

下面我们来看一下配置了黑洞路由的情况。首先在防火墙上配置一条目的地址是 NAT 地址池地址的黑洞路由，为了避免这条黑洞路由影响其他业务，我们将掩码配置成 32 位，精确匹配 202.1.1.10 这个地址：

```
[FW] ip route-static 202.1.1.10 32 NULL 0
```

然后在防火墙的 GE0/0/1 接口上开启抓包，在公网 PC 上执行 **ping 202.1.1.10 -c 1** 命令，还是只发送一个 ping 报文，查看抓包信息：



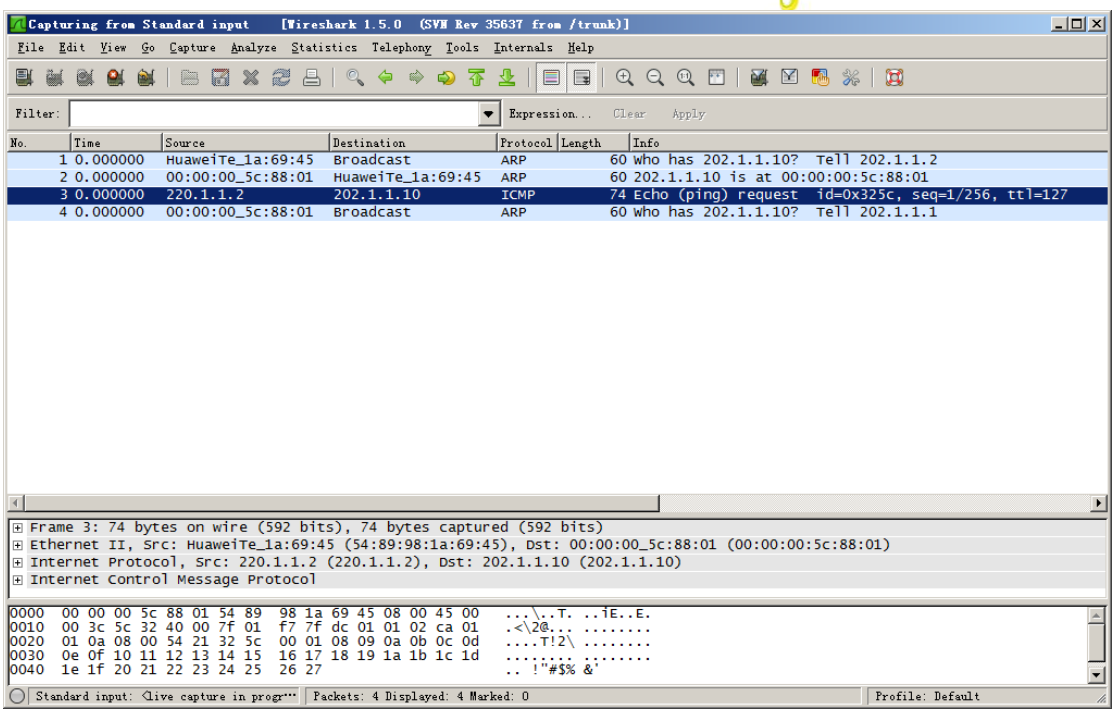
只抓到了一个 ICMP 报文，说明防火墙收到路由器发送过来的报文后，匹配到了黑洞路由，直接将报文丢弃了。此时就不会在防火墙和路由器之间产生路由环路，即使防火墙收到再多的同类报文，都会送到黑洞中，一去不复返。并且，这条黑洞路由不会影响正常业务，私网 PC 还是可以正常访问公网 Server。

到这里大家可能会问了，没有配置黑洞路由时，报文最终也会被丢弃，没啥问题啊？上面我们只是用了一个 ping 报文来演示这个过程，试想一下，如果公网上的捣乱分子利用成千上万的 PC 主动向 NAT 地址池地址发起大量访问，无数的报文就会在防火墙和路由器之间循环转发，占用链路带宽资源，同时防火墙和路由器将会消耗大量的系统资源来处理这些报文，就可能导致无法处理正常的业务。

所以，**当防火墙上 NAT 地址池地址和公网接口地址不在同一网段时，必须配置黑洞路由，避免在防火墙和路由器之间产生路由环路**。

那么如果 NAT 地址池地址和公网接口地址在同一网段时，还会有这个问题吗？我们再来验证一下。

首先，将防火墙和路由器互联的两个接口的掩码修改为 24 位，这样接口地址和 NAT 地址池地址就在同一网段了，然后去掉黑洞路由的配置，接下来在防火墙的 GE0/0/1 接口抓包，在公网 PC 执行 **ping 202.1.1.10 -c 1** 命令，查看抓包信息：



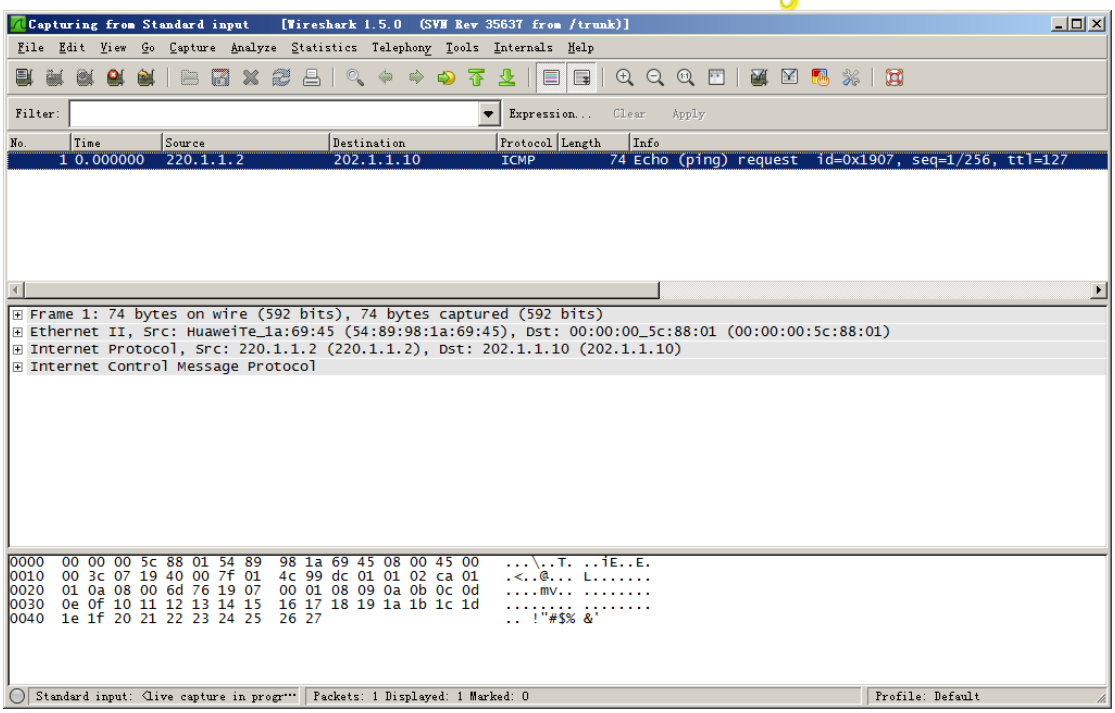
我们发现，只抓到了三个 ARP 报文和一个 ICMP 报文，公网 PC 访问 NAT 地址池地址的报文没有在防火墙和路由之间相互转发。梳理整个过程：

1. 路由器收到公网 PC 访问 NAT 地址池地址的报文后，发现目的地址属于自己的直连网段，发送 ARP 请求，防火墙会回应这个 ARP 请求，前两个 ARP 报文就是来完成了这一交互过程的。然后路由器使用防火墙告知的 MAC 地址封装报文，发送至防火墙。

2. 防火墙收到报文后，发现报文的目的地址和自己的 GE0/0/1 接口在同一网段，直接发送 ARP 请求报文（第三个 ARP 报文），寻找该地址的 MAC 地址（防火墙依然没有意识到该报文的目的地址是自己的 NAT 地址池地址）。但是网络中其它设备都没有配置这个地址，肯定就不会回应，最终防火墙将报文丢弃。

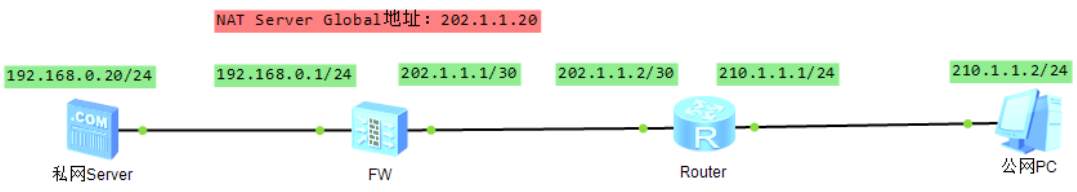
所以说，在这种情况下不会产生路由环路。但是如果公网上的捣乱分子发起大量访问时，防火墙将发送大量的 ARP 请求报文，也会消耗系统资源。所以，**当防火墙上 NAT 地址池地址和公网接口地址在同一网段时，建议也配置黑洞路由，避免防火墙发送 ARP 请求报文，节省防火墙的系统资源。**

下面是配置黑洞路由后的抓包信息，可以看到，防火墙没有再发送 ARP 请求报文：



还有一种极端情况，我们配置源 NAT 时，可以直接把公网接口地址作为转换后地址（easy-ip 方式），也可以把公网接口地址配置成地址池地址。这样，NAT 转换使用的地址和公网接口地址就是同一个地址了。在这种情况下，需要配置黑洞路由吗？我们来分析一下整个流程：防火墙收到公网 PC 的报文后，**发现是访问自身的报文**，这时候就取决于公网接口所属安全区域和 Local 安全区域之间的安全策略，安全策略允许通过，就处理；安全策略不允许通过，就丢弃。不会产生路由环路，也不需要配置黑洞路由。

看到这里，聪明的小伙伴肯定会问，NAT Server 有没有这个问题啊？强叔告诉大家，NAT Server 也存在路由环路的问题，不过发生路由环路的前提条件比较特殊，要看 NAT Server 是怎样配置的。下面是一个典型的 NAT Server 组网环境，我们先来看一下 NAT Server 的 Global 地址和公网接口地址不在同一网段的情况。



如果我们在防火墙上配置了一条粗犷型的 NAT Server，将私网 Server 全部发布到公网，如：

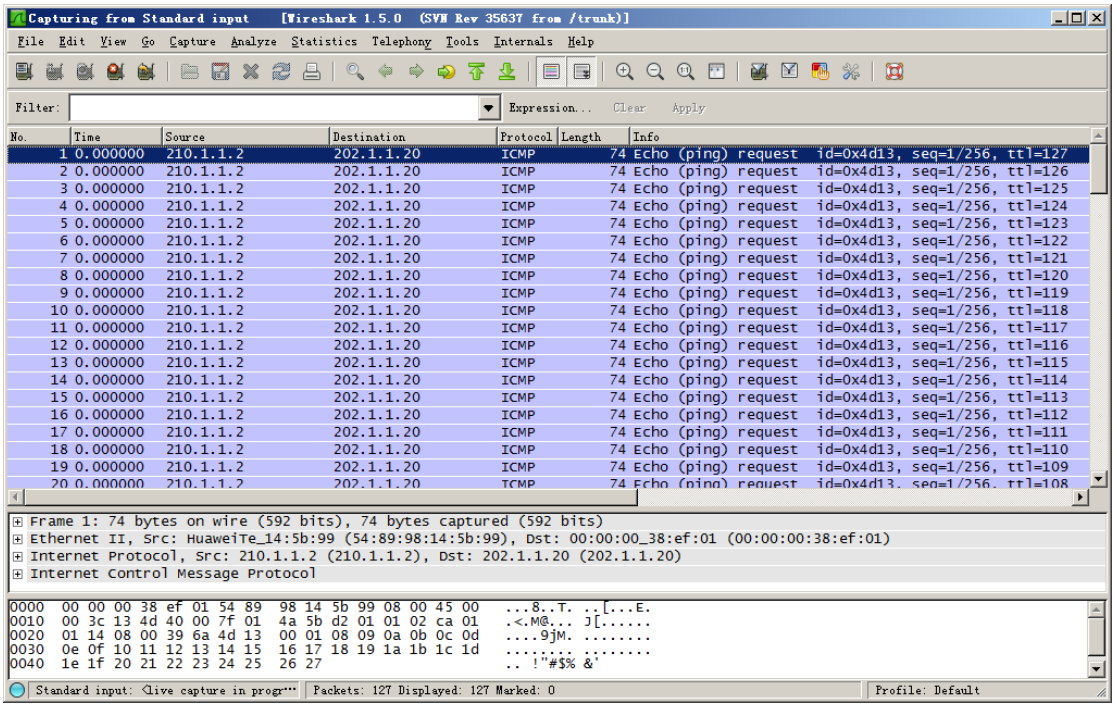
```
[FW] nat server global 202.1.1.20 inside 192.168.0.20
```

公网 PC 访问 202.1.1.20 的报文，目的地址都会被转换成 192.168.0.20，然后发送给私网 Server，这个时候自然不会产生路由环路。

但是如果我们配置了一条精细化的 NAT Server，只把特定的端口发布到公网上，如：

```
[FW] nat server protocol tcp 202.1.1.20 80 inside 192.168.0.20 80
```

此时如果公网 PC 不按常理出牌，没有访问 202.1.1.20 的 80 端口，而是使用 **ping** 命令访问 202.1.1.20，防火墙收到该报文后，既无法匹配 Server-map 表，也无法匹配会话表，就只能查找路由转发，从 GE0/0/1 接口送出去。而路由器收到报文后，还是要送到防火墙，这样依然会产生路由环路：



所以，**当防火墙上配置了特定协议和端口的 NAT Server 并且 NAT Server 的 Global 地址和公网接口地址不在同一网段时，必须配置黑洞路由，避免在防火墙和路由器之间产生路由环路。**

如果 NAT Server 的 Global 地址和公网接口地址在同一网段，防火墙收到公网 PC 的 ping 报文后，会发送 ARP 请求报文，这个过程就和前面讲过的 NAT 的情况是一样的。同理，**当防火墙上配置了特定协议和端口的 NAT Server 并且 NAT Server 的 Global 地址和公网接口地址在同一网段时，建议也配置黑洞路由，避免防火墙发送 ARP 请求报文，节省防火墙的系统资源。**

同样，我们配置 NAT Server 时，也可以把公网接口地址配置成 Global 地址。此时，防火墙收到公网 PC 的报文后，如果能匹配上 Server-map 表，就转换目的地址，然后转发到私网；如果不能匹配上 Server-map 表，就会**认为是访问自身的报文**，由公网接口所属安全区域和 Local 安全区域之间的安全策略决定如何处理，不会产生路由环路，也不需要配置黑洞路由。讲到这里，相信大家一定明白了配置黑洞路由的原因，是不是感觉内功又提升了啊~~我们再总结一下：

对于源 NAT 来说：

- 如果 NAT 地址池地址与公网接口地址不在同一网段，必须配置黑洞路由。
- 如果 NAT 地址池地址与公网接口地址在同一网段，建议也配置黑洞路由。

对于配置了特定协议和端口的 NAT Server 来说：

- 如果 NAT Server 的 Global 地址与公网接口地址不在同一网段，必须配置黑洞路由。
- 如果 NAT Server 的 Global 地址与公网接口地址在同一网段，建议也配置黑洞路由。

最后，大家再来跟强叔默念一下本篇内功心法的口诀：地址不在同网段，报文无奈来回走，黑洞路由必须配，防止环路显身手；地址虽在同网段，但是会发 ARP 请求，黑洞路由也配上，节省资源不用愁！

强叔提问

在《NAT Server 三十二字真言（下篇）》中，强叔提到了 **zone** 参数，请大家试着分析一下，如果配置了带有 **zone** 参数的 NAT Server 后，是否也需要配置黑洞路由呢？

扩展阅读

除了防止路由环路、节省设备的系统资源，其实黑洞路由还有一个作用，那就是在防火墙上引入到 OSPF 中，发布给路由器。

我们知道，当 NAT 地址池地址或 NAT Server 的 Global 地址与防火墙和路由器互联接口的地址不在同一网段时，需要在路由器上配置到 NAT 地址池地址或 NAT Server 的 Global 地址的静态路由，保证路由器可以把去往 NAT 地址池地址或 NAT Server 的 Global 地址的报文发送到防火墙。

如果防火墙和路由器之间运行 OSPF 协议，那么就可以通过 OSPF 协议来学习路由，减少手动配置的工作量。但是 NAT 地址池地址和 NAT Server 的 Global 地址不同于接口地址，无法在 OSPF 中通过 network 的方式发布出去，那么路由器如何才能学习到路由呢？

此时就可以通过在防火墙的 OSPF 中引入静态路由的方式，把黑洞路由引入到 OSPF 中，然后通过 OSPF 发布给路由器。这样，路由器就知道了去往 NAT 地址池地址或 NAT Server 的 Global 地址的报文都要发送到防火墙上（注意，是发送到防火墙，而不是发送到黑洞中）。

以 NAT Server 的组网为例，NAT Server 的 Global 地址和公网接口地址不在同一网段，防火墙和路由器都运行 OSPF 协议，在防火墙上的 OSPF 中引入静态路由：

```
#
```

```
ospf 100
 import-route static
 area 0.0.0.0
  network 202.1.1.0 0.0.0.3
#
```

这时路由器就可以学习到去往 NAT Server 的 Global 地址的路由：

```
[Router]display ip routing-table
Route Flags: R - relay, D - download to fib
------------------------------------------------------------------------------
Routing Tables: Public
         Destinations : 7        Routes : 7

Destination/Mask    Proto   Pre  Cost      Flags NextHop         Interface

      127.0.0.0/8   Direct  0    0           D   127.0.0.1       InLoopBack0
      127.0.0.1/32  Direct  0    0           D   127.0.0.1       InLoopBack0
      202.1.1.0/30  Direct  0    0           D   202.1.1.2       Ethernet0/0/0
      202.1.1.2/32  Direct  0    0           D   127.0.0.1       Ethernet0/0/0
     202.1.1.20/32  O_ASE   150  1           D   202.1.1.1       Ethernet0/0/0
      210.1.1.0/24  Direct  0    0           D   210.1.1.1       Ethernet0/0/1
      210.1.1.1/32  Direct  0    0           D   127.0.0.1       Ethernet0/0/1
```

Firewall

本期策划：徐慧洋

本期作者：王蕾、白杰、刘水、
韩姣、闫广辉、余杨

美术编辑：潘艺从

版权所有
©华为技术有限公司2014
保留一切权利

关于华为防火墙，你还想了解更多吗？请点击【强叔侃墙】汇总贴

如有任何问题，请留言至xuhuiyang.xu@huawei.com