

通过缓存一致性检查方法和灾难解析系统优化来抵御 DNS 缓存中毒

杨天乐, 谌颖, 王海涛

(中广电移动网络有限公司, 北京 100020)

摘要: 域名系统 (DNS, Domain Name System) 是互联网的核心组成部分, 但由于分布式和缓存特性, DNS 容易受到各种攻击, 尤其是缓存中毒。随着随机端口号和随机事务 ID 的使用, 缓存中毒的概率有所降低, 但是近几年随着 DNS Forwarder 分片整理和侧信道攻击的出现, 缓存中毒的概率又有加大的趋势。为应对新出现的缓存中毒手段, 本文提出了 3C (Cache Consistency Checking) 方法, 通过检查 DNS 缓存和权威查询结果是否一致来判断 DNS 是否发生缓存中毒, 缓存中毒后切换到容灾解析系统进行解析。同时为了加快比对速度和 DNS 查询速度, 隔离缓存中毒对容灾解析系统的影响, 本文使用了本地顶级域权威镜像查询系统。实验证明 3C 方法能准确检测缓存中毒, 而本地权威镜像查询系统可以大大提高 3C 的比对效率。与传统 DNS 相比, 集成 3C 方法和本地顶级域权威镜像查询系统的 DNS 查询更快, 提升了 DNS 的性能和安全性。

关键词: 域名系统; 缓存中毒; 缓存一致性检查; 容灾解析; 分片整理攻击

本文著录格式: 杨天乐, 谌颖, 王海涛. 通过缓存一致性检查方法和灾难解析系统优化来抵御 DNS 缓存中毒[J]. 新一代信息技术, 2023, 6(16): 30-36

中图分类号: TN915; TP393

文献标识码: A

Mitigating DNS Cache Poisoning Through Cache Consistency Checking Method and Disaster Resolution System Optimization

YANG Tian-le, CHEN Ying, WANG Hai-tao

(China Broadnet Mobile Network Co., Ltd, Beijing 100020)

Abstract: DNS (Domain Name System) is a fundamental component of the internet. However, due to its distributed and caching nature, DNS is vulnerable to various attacks, especially cache poisoning. While the use of random port numbers and transaction IDs has reduced the likelihood of cache poisoning, recent developments such as DNS Forwarder fragmentation and side-channel attacks have increased the risk. To address these emerging cache poisoning techniques, this paper introduces the 3C (Cache Consistency Checking) method. It assesses whether DNS cache and authoritative query results align, detecting cache poisoning and subsequently switching to a disaster recovery resolution system for resolution. To enhance comparison speed and DNS query efficiency, and to isolate the impact of cache poisoning on the disaster recovery resolution system, the paper employs a local top-level domain authoritative mirror query system. Experimental results demonstrate the accuracy of the 3C method in detecting cache poisoning, while the local authoritative mirror query system significantly enhances 3C's comparative efficiency. Compared to traditional DNS, the integration of the 3C method and the local top-level mirror system results in faster DNS queries, thereby improving DNS performance and security.

Key words: domain name system; cache poisoning; cache consistency checking; disaster recovery resolution; fragmentation attack

Citation: YANG Tian-le, CHEN Ying, WANG Hai-tao. Mitigating DNS Cache Poisoning Through Cache Consistency Checking Method and Disaster Resolution System Optimization[J]. New Generation of Information Technology, 2023, 6(16): 30-36

1 引言

随着互联网的快速发展,域名系统(Domain Name System, DNS)作为关键的网络基础设施,扮演着将域名解析为IP地址的重要角色。然而,由于其开放性和分布式特性,DNS系统也面临着各种安全威胁,其中包括DNS缓存中毒攻击。这些攻击不仅会干扰互联网的正常运行,还可能导致用户数据泄露、劫持敏感信息和恶意篡改网站内容等危害。臭名昭著的Kaminsky攻击提高了全世界对DNS缓存中毒严重性的认识,并促使部署了几种协议级防御机制。针对DNS缓存中毒,已有一些方法和技术被提出来,用于检测和防范,其中DNSSEC协议、防火墙和入侵检测系统等被广泛应用。然而,近年来出现了一系列新的漏洞^[1-3],这些漏洞重新引发了缓存中毒和DNS劫持的威胁。

尤其是针对DNS Forwarder的分片整理攻击是一种相对较新且更具挑战性的攻击方式^[4],它利用了DNS响应数据包的分片特性来绕过现有防御措施,将恶意响应数据缓存到DNS Forwarder中,进而导致缓存中毒。这种攻击不仅对普通用户构成威胁,还可能影响企业和组织的网络安全。

为了有效应对这一攻击威胁,本文提出了一种新颖的防范方法,即基于缓存一致性检测方法的3C(Cache Consistency Checking)防御策略。该方法通过对递归查询的响应数据进行缓存一致性检查,及时发现并防止缓存中毒情况发生。相比传统方法,3C方法不依赖于DNSSEC的部署,也不需要响应数据进行截断或改写,从而减少了额外的计算和带宽开销。

本文将详细介绍3C方法的原理与实现,并在实验环境中对其进行评估。通过对比实验结果,我们验证了该方法的有效性和可行性。此外,本文还对3C方法的优势进行深入探讨,阐述了其在防范DNS Forwarder的分片整理攻击方面的应用前景。

2 DNS缓存中毒攻击相关工作

2.1 DNS缓存中毒攻击的背景和最新发展趋势

DNS缓存中毒攻击是当前互联网安全领域面临的重要挑战之一,许多研究者和安全专家已经提出了各种方法和技术来应对这一威胁,DNS缓存中毒攻击原理如图1所示。其中,伪造攻击的目标是制造一个恶意的DNS响应,并欺骗解析器接受它。攻击者利用伪造的数据包中的字段与DNS查询匹配,例如问题部分、DNS事务ID、源/目的地址和端口号。如果攻击者在合法响应到达之前,伪造一个正确的数据包,那么

非法响应可能会被解析器接受,攻击成功。最具影响力的伪造攻击案例是2008年的Kaminsky攻击,它影响了几乎所有涉及用于DNS的软件。为了减少伪造攻击的成功率,现有的防御措施包括增加DNS查询报文的随机性,如使用随机的临时端口号和DNS事务ID,并采用0×20编码来混合问题中名称的大小写拼写。

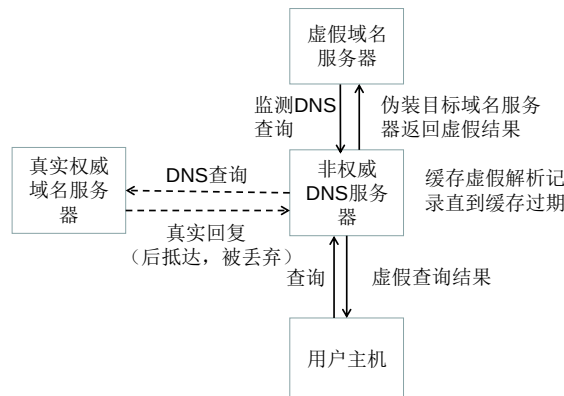


图1 DNS缓存中毒攻击原理

另一种DNS缓存面临的攻击是碎片整理攻击,它利用分片化DNS响应数据包的特性,绕过了基于随机化的伪造攻击防御。攻击者通过制造恶意的DNS记录,将受害域名发送DNS查询,并使来自权威服务器的响应被强制分片。然后在递归解析器上,攻击者将合法的第一个片段与恶意的第二个片段重新组装,生成恶意的DNS响应,并成功投毒缓存。为了解决碎片整理攻击,目前已经提出了两种方法:一种是降低递归解析器和授权服务器之间的路径最大传输单元(Path Maximum Transmission Unit, PMTU),称为基于PMTU的碎片整理攻击;另一种是发送DNSSEC查询以扩大DNS响应的大小,达到最大传输单元(Maximum Transmission Unit, MTU)的限制,并使其被分片,称为基于DNSSEC的碎片整理攻击。

传统的DNS缓存中毒攻击,除了2008年的Kaminsky攻击影响范围比较大外,之后随着随机化端口和随机化交易ID的引入,进行缓存中毒的概率只有1/232。但是最近出现的针对DNS Forwarder的分片整理攻击^[4]和通过ICMP包探测端口的技术^[1]将这一概率大大提升了。

针对DNS Forwarder的分片整理攻击的具体步骤如下:

(1)攻击者预测IPID:攻击者需要预测目标DNS Forwarder使用的IP数据包的IPID(即TCP的分包标识,在IPv4包中是首部的16位IP标识,在IPv6数据包中是分段扩展头中的32位分段标识),这是攻击的关键之一。攻击者可以通过发送多个探测数据包,并观察IPID的变化规律,来预测下一个数据包的IPID。

(2)构造第2个分片:攻击者构造响应数据包的第2个分片,并在该分片中包含一条伪造的A记录,将目标域名映射到攻击者控制的恶意IP地址。这个第2个分片是不完整的,它不包含DNS头部和DNS查询部分,仅包含响应数据。

(3)发起DNS查询请求:攻击者发起DNS查询请求,查询目标域名(例如:http://a.attacker.com)的IP地址。DNS Forwarder将该查询转发给上游的递归解析器。

(4)权威域名解析器返回恶意响应:由于权威域名解析器是被攻击者控制的,它会返回一个带有大量CNAME记录的响应数据包给递归解析器。这些CNAME记录将响应数据包填充得很大。

(5)分片数据包重组:递归解析器收到响应数据包后,将其分片发送给DNS Forwarder。DNS Forwarder接收到第1个分片时,会将其与之前缓存的第2个分片重新组合成完整的数据包,这里能够重新组装是因为第2个分片中的IPID是预测过的,能和第1个合法的分片匹配上。由于DNS Forwarder不会对数据包进行检查,它会直接将这个结果转发给DNS客户端。

(6)DNS缓存中毒:DNS客户端收到恶意响应后,将恶意IP地址缓存起来。以后对相同域名的查询都将返回恶意IP地址,从而导致DNS缓存中毒。

以上是针对DNS Forwarder的分片整理攻击的详细步骤,攻击者利用分片数据包的特性来欺骗DNS Forwarder,并成功使其缓存恶意IP地址。

2.2 防范DNS缓存中毒攻击的措施

为了防范DNS缓存中毒攻击,经常采用下面一些方法:

(1)DNSSEC(Domain Name System Security Extensions)是一种被广泛用于防范DNS缓存中毒的安全扩展机制。它通过为DNS数据包添加数字签名来验证响应数据的真实性,从而防止恶意的数据篡改。DNSSEC可以提供数据完整性和认证,有效地阻止了DNS数据包被篡改的可能性^[5]。然而,DNSSEC的部署和配置相对复杂,需要域名注册商和DNS服务器的支持,因此在实际应用中部署率较低。

(2)防火墙和入侵检测系统是常见的网络安全设备,用于监控和过滤网络流量,防止未经授权的访问和攻击^[6-7]。在防范DNS缓存中毒方面,防火墙可以配置规则来限制对DNS服务器的访问,从而防止一些恶意请求的传入。入侵检测系统可以检测和识别异常的DNS流量和行为,从而及时发现可能的缓存中毒攻击^[8]。然而,防火墙和入侵检测系统往往难以识别针对DNS Forwarder的分片整理攻击,因为这种攻击利用

了数据包的分片特性,绕过了数据包的完整性检查。

(3)DNS Cookies是一种解决DNS安全性问题的技术,引自RFC7873。它旨在应对DNSSEC无法解决的缓存中毒和欺骗性DNS放大攻击等问题。DNS Cookies实现简单,但安全性与使用TCP类似,相较于DNSSEC更易部署。客户端发送DNS请求时携带Cookie选项,其中包含IP地址和secret的hash值。服务器使用客户端的IP和secret验证Cookie,确保一致性。DNS Cookies通过简单的机制提升DNS安全性,为防御缓存中毒等攻击提供有效补充。相关研究结果显示,使用DNS Cookies的服务器占比不到30%,递归客户端中使用Cookie的比例不到10%^[9-10]。

(4)DNS请求频率限制是一种有效的防范DNS缓存中毒的方法。它通过限制特定IP地址在一定时间内对DNS服务器的请求频率,防止大量的恶意请求导致DNS服务器的过载和缓存中毒。这种方法对于一些恶意的DDoS攻击和频繁请求的恶意行为有一定的抵御能力。然而,针对分片整理攻击,攻击者可以通过控制合法的DNS服务器进行缓存投毒,绕过请求频率限制。

(5)3C方法是本文提出的一种新颖的防范DNS缓存中毒的策略。该方法通过对DNS Forwarder的响应数据进行缓存一致性检查,及时发现并防止缓存中毒情况。与其他方法相比,3C方法不依赖于DNSSEC的部署,也不需要响应数据进行截断或改写,从而减少了额外的计算和带宽开销。通过引入3C方法,我们可以更好地防范针对DNS Forwarder的分片整理攻击,提高DNS系统的安全性和稳定性。

综上所述,当前已有许多方法和技术可用于防范DNS缓存中毒,每种方法都有其优势和局限性。通过综合使用多种方法,并结合新颖的3C方法,我们可以构建更加强大和有效的DNS缓存中毒防范系统,为互联网用户和组织提供更可靠的网络安全保障。同时,我们还需要不断探索和创新,保持对互联网安全领域的关注,不断完善DNS缓存中毒防御策略。

3 3C方法的原理与实现

3C方法是本文提出的一种新颖的防范DNS缓存中毒的策略。该方法主要通过对DNS缓存的响应数据进行缓存一致性检查,以及实时对比DNS响应与权威查询的结果,及时发现并防止缓存中毒情况。

3.1 3C方法的原理

3C方法的核心原理是对DNS Forwarder的缓存数据进行监测和比对。在普通的DNS响应中,DNS缓存

服务器会将响应数据包含有关域名的IP地址记录缓存起来,以便在用户再次查询相同域名时能够快速响应,减少DNS解析时间。然而,当DNS Forwarder受到针对分片整理攻击时,恶意响应数据可能被缓存到DNS Forwarder中,进而导致缓存中毒。

3C方法首先在DNS Forwarder中建立一个缓存数据库,记录所有已缓存的域名和对应的IP地址记录。然后,同时对权威查询的结果进行实时监测。当DNS Forwarder接收到用户的DNS查询请求后,它会先查找缓存数据库,如果查询结果在缓存中存在,则直接返回给用户;如果缓存中不存在,则向权威域名解析器

发起查询请求。

在获得来自权威域名解析器的响应后,3C方法会将该响应与缓存数据库中对应域名的缓存数据进行比对。如果两者的IP地址记录一致,说明响应数据是合法的,并且没有遭到缓存中毒攻击,那么DNS Forwarder将该响应数据缓存,并返回给用户。如果发现两者的IP地址记录不一致,那么就有可能发生了缓存中毒攻击。在这种情况下,3C方法将立即警报,并停止将恶意响应数据缓存,从而保护用户的安全。

缓存比对方法主要有以下几种,这几种缓存一致性检查常用方法的优缺点对比如表1所示。

表1 缓存一致性检查常用方法优缺点对比

检查方法	优点	缺点
哈希校验方法	检测效率比较高,提供更高的数据完整性,可以有效检测缓存数据的篡改	对于解析记录结果存在细微区别的时候无效。可能存在哈希冲突,导致误判
解析记录比对方法	资源开销比哈希校验方要高,但是比对的结果更加准确	需要一条条的解析出来查询域名的解析结果,资源消耗较大
区块链技术	具有不可篡改性和去中心化特性,提供高度可信的数据存储和验证。可以实时记录和验证缓存数据,防止数据篡改	引入了复杂性和资源开销,包括共识算法的计算和区块链维护的成本。对于高频的DNS查询,可能导致区块链的容量和速度问题
冗余数据对比	可以通过多个独立的缓存节点进行数据对比,增加了缓存一致性的可信度。不需要依赖复杂的算法,易于实现和部署	需要维护多个独立的缓存节点,增加了资源和管理成本。对于大规模系统,可能会导致网络延迟和数据同步问题

(1)哈希校验方法:对DNS缓存和权威查询缓存中的数据进行哈希计算,将计算得到的哈希值进行比对,若存在不一致,则视为可能发生了缓存中毒。

(2)解析记录比对方法:通过对DNS递归缓存的解析结果和权威查询返回的解析结果进行比对,检测是否存在解析不一致的情况,从而判断是否出现缓存中毒。

(3)区块链技术:利用区块链的不可篡改性和分布式特性,将DNS查询和缓存更新的记录存储在区块链上,通过比对区块链中的数据来验证缓存的一致性。

(4)冗余数据对比:将同一DNS查询结果同时缓存到多个独立的DNS缓存系统中,然后对这些缓存数据进行对比,如果存在不一致,就表明可能发生了缓存中毒。

3.2 3C方法的实现

为了应对系统复杂度和计算复杂度,我们综合使用了哈希校验方法和解析记录比对方法来实现3C方法。首先,在获取权威域名解析器的响应后,我们对响应(answer)部分进行哈希运算,然后对解析值进行比对。若哈希运算结果和解析对比结果都不一致,则表明可能发生了缓存中毒,进而触发警告提示。若只

有一个检查结果不一致,则将查询的域名纳入重点监测列表。当结果比对一致时,表示DNS查询系统正常运行。具体而言,3C方法的实现包括以下步骤:

(1)建立缓存数据库:建立一个缓存数据库,用于记录已缓存的域名与对应的IP地址记录。这个数据库可以采用关系型数据库或者NoSQL数据库进行存储。

(2)监测DNS查询请求:当缓存服务器接收到用户的DNS查询请求时,首先查找缓存数据库。如果查询结果在缓存中存在,则直接返回给用户;如果缓存中不存在,则向权威域名解析器发起查询请求。

(3)监测权威查询响应:在收到来自权威域名解析器的响应后,将该响应与缓存数据库中对应域名的缓存数据进行比对。通过哈希校验方法和解析记录比对方法来确保数据的完整性和一致性。

(4)缓存数据更新与警报:若发现两者的IP地址记录一致,说明响应数据是合法的,没有发生缓存中毒攻击,那么缓存服务器将该响应数据缓存并返回给用户。若发现两者的IP地址记录不一致,可能发生了缓存中毒攻击,3C方法将立即发出警报,并停止将恶意响应数据缓存,以保护用户的安全。

(5)缓存一致性维护:定期对缓存数据库中的数据进行更新和清理,确保缓存数据的准确性和一致

性。同时,引入缓存过期机制,根据 TTL(Time To Live)自动清理过期的缓存数据。

通过以上步骤,3C方法可以有效地防范针对 DNS Forwarder 的分片整理攻击和侧信道攻击等缓存中毒攻击手段,从而维护 DNS 解析的安全和稳定性。

3.3 3C 方法的优势与不足

3C方法具有多方面的优势。首先,该方法能够实时监测 DNS 响应数据与权威查询结果的一致性,一旦发现缓存中毒攻击,能够及时警报,有效减少恶意响应数据的传播;其次,3C方法在实施时具备灵活性和可扩展性,能够根据实际情况选择不同的比对方式和数据存储方式,因此具有较高的适应性。此外,该方法可以与其他防范 DNS 缓存中毒的策略相结合,形成综合的防护体系,从而提高 DNS 解析的安全性。

然而,3C方法也存在一些挑战和不足。首先,实时监测 DNS 响应与权威查询的一致性需要额外的计算和存储开销,这可能会对 DNS 解析的性能产生一定影响;其次,尽管该方法可以有效防范缓存中毒攻击,但如果权威服务器被攻击,也无法预防;最后,为保持缓存一致性,缓存数据库的及时更新和维护非常关键,必须保证数据库的准确性和时效性。

为了降低 3C 方法存在的不足,我们通过部署本地顶级权威镜像查询系统来提高 3C 比对缓存的效率,同时提高容灾解析系统的安全性。

4 应对 DNS 缓存中毒的容灾解析系统与优化

在本研究中,我们通过 3C 方法来防范缓存中毒攻击,而 3C 方法有一个比较大的性能瓶颈,就是需要经常去做本地缓存解析记录和通过权威查询获得的解析记录的比对。而权威解析记录的查询是一个比较耗费时间的过程^[10-11]。所以为了提升 3C 方法的效率,同时在启动容灾递归解析系统的时候,启动一个本地顶级权威镜像查询系统提高比对速度。容灾解析系统为检测到 DNS 缓存中毒后提供了有效的应对措施。该容灾解析系统包含两个主要部分,分别是应急缓存查询系统和本地顶级权威镜像查询系统,以提供安全可靠的 DNS 解析服务,如图 2 所示的 3C 系统结构。

首先,我们引入了应急缓存查询系统,该系统是一组备用服务器,提前加载了经过验证的、没有缓存中毒的 DNS 查询缓存。当 3C 方法检测到 DNS 缓存中毒时,即发现 DNS 缓存的响应数据与权威数据不一致时,我们立即将不一致的 DNS 查询转移到容灾解析服务器上。通过这一措施,我们确保用户的 DNS 查询得

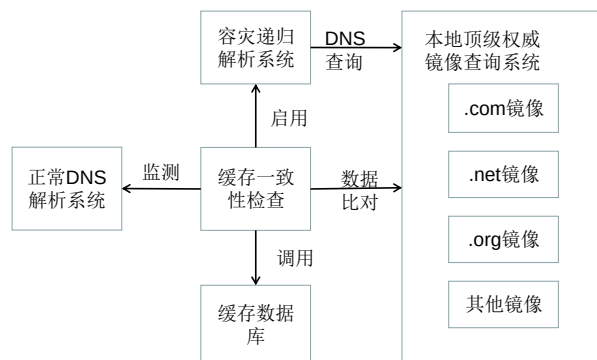


图 2 3C 系统结构

到及时的响应,同时避免了缓存中毒对 DNS 解析服务的不良影响。

其次,我们在容灾解析系统中引入了本地顶级权威镜像查询系统,该系统通过从集中区数据服务(CZDS)下载 .com 和 .net 等顶级域名数据,并记录下载速度和下载所需时间,将这些数据加载到本地的镜像缓存中,以提供本地的权威数据查询。在 DNS 解析时,我们可以更快地获取到 .com 和 .net 等顶级域名的权威数据,从而加速缓存数据与权威数据的比对过程。通过这一优化措施,我们成功降低了缓存数据与权威数据比对的延迟,提高了系统的整体性能。

综上所述,我们通过引入应急缓存查询系统和本地顶级权威镜像查询系统,有效地防范了 DNS 缓存中毒攻击,并优化了 3C 方法中的数据比对速度。这些措施为 DNS 解析服务的安全性和性能提供了坚实的保障,使得用户在使用 DNS 服务时能够获得更加可靠和快速的解析体验。然而,随着攻击技术的不断进化,我们还需要不断改进和优化防御措施,以应对未来可能出现的新型 DNS 缓存中毒攻击。因此,我们对未来的研究方向充满期待,并将持续致力于提高 DNS 系统的安全性和性能。

5 实验与评估

为了验证 3C 方法的有效性和性能,我们进行了一系列实验,并对其结果进行了详细评估。在实验中,我们模拟了针对 DNS Forwarder 的分片整理攻击,并使用 3C 方法进行防御。以下是实验的具体步骤和评估结果。

5.1 实验步骤

(1)环境搭建:我们在实验中搭建了一个模拟的 DNS 解析环境,包括 DNS Forwarder、权威域名解析器和 DNS 客户端。我们使用开源软件 Bind 和 Unbound 来搭建 DNS 服务器,并在本地网络中部署这些服务器。

(2)分片整理攻击模拟:为了模拟分片整理攻击,我们利用Python脚本构造了恶意的DNS响应数据包,并向DNS Forwarder发送这些响应数据包。这些响应数据包中包含伪造的A记录和CNAME记录,指向恶意IP地址。同时,我们使用Scapy库来控制数据包的分片,并预测IPID字段的值,以实现分片整理攻击。

(3)3C方法实现:我们在DNS Forwarder中实现了3C方法。具体实现包括建立缓存数据库、监测DNS查询请求和权威查询响应、比对响应数据与缓存数据的一致性,以及实时警报机制。

(4)实验运行:我们运行了多组实验,模拟不同类型的分片整理攻击,并观察3C方法的防御效果。同时,我们记录了实验中的各项数据,如缓存命中率、警报次数等。

5.2 实验结果

防御效果:实验结果表明,3C方法能够有效防御针对DNS Forwarder的分片整理攻击。在攻击模拟下,我们观察到3C方法成功检测并警报了恶意响应数据包,并停止将其缓存,从而保护了DNS解析的安全。

性能评估:我们对3C方法的性能进行了评估。实验结果显示,3C方法对DNS Forwarder的性能影响较小。缓存数据库的建立和维护耗时较短,并且实时监控和比对过程所需的计算开销也较小。

在对本地顶级权威镜像查询系统的验证测试中,我们选择从CZDS下载最新的.com和.net顶级域名数据进行分析。.com区域文件gz压缩文件共4.1 GB,解压后的文本文件共22 GB,而.net区域文件gz压缩文件共385 MB,解压后的文本文件为1.9 GB,“.com”下有157 844 204个域名,“.net”下有13 494 414个域名。如图3所示,.com域数据的平均下载时间为6 min 28 s,.net域数据的平均下载时间为42 s。如图4所示,使用Bind9对.com和.net顶级域文件进行多次加载,并分析了加载测试时间.com冷数据加载平均时间为56 min。如图5所示,.com重新加载区域文件的平均时间为59 min 17 s。

灵活性与可扩展性:3C方法在实验中表现出较高的灵活性和可扩展性。我们可以根据实际情况选择不同的比对方式和数据存储方式,并对警报机制进行调整。

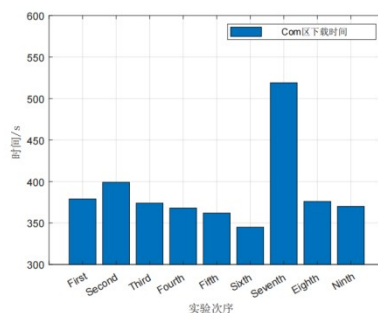


图3 Com 区数据下载时间

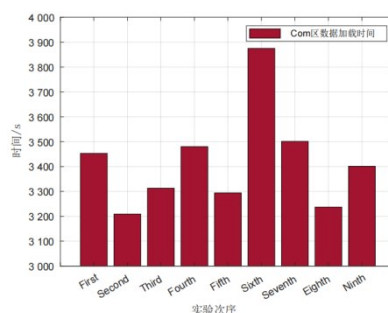


图4 Com 区数据加载时间

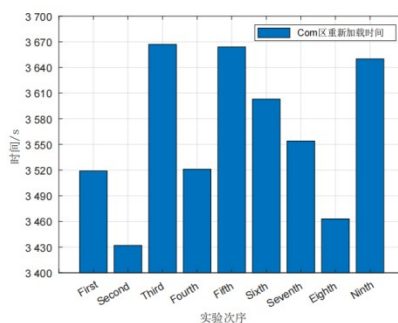


图5 Com 区数据重新加载时间

与其他方法的对比:我们将3C方法与其他常用的DNS缓存中毒防御方法进行了对比。如图6所示,从图中可以看出来,使用了3C系统之后,NXDOMAIN域名的数量和比例比不使用的时候,都有所下降。实验结果显示,3C方法在防御分片整理攻击方面表现较为优秀,并在综合性能上取得了一定的优势。3C系统对权威解析时间的提升如表2所示。

综合来看,3C方法在实验中表现出了良好的防御效果和较高的性能。然而,仍然需要进一步的研究和优化,以提高其实际应用的效果和性能。未来,我们可以进一步探索更多的缓存一致性检查机制,引入机

表2 3C系统对权威解析时间的提升

表项	正常解析成功率/%	容灾解析成功率/%	正常权威平均解析时间/ms	容灾权威平均解析时间/ms
正常 1K 域名	100	100	15.055	14.807
随机 1K 域名	99.9	100	68.819	5.232

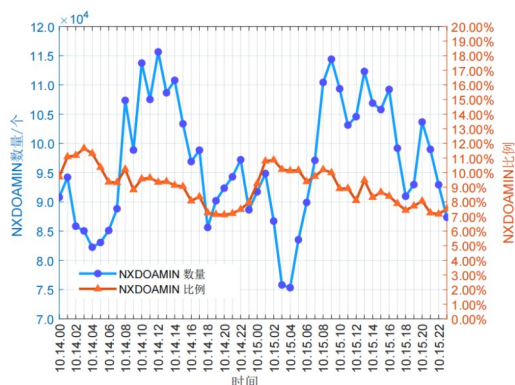


图6 NXDOMAIN域名的数量和比例

器学习等技术来提高检测的准确性和效率。同时,也可以进一步研究其他类型的DNS缓存中毒攻击,构建更为全面的DNS安全防护体系。通过持续努力,我们相信3C方法能够为保护DNS解析的安全和稳定作出更大的贡献。

6 结论

本文探讨了当前DNS缓存攻击的现状,重点分析了DNS Forwarder的分片整理攻击以及DNS侧道攻击等方面的攻击方式。为了防范这些新型的DNS缓存攻击,本文提出了采用3C方法的防御策略,该方法通过实时比对缓存数据和权威数据的一致性,及时发现和阻止恶意响应数据的缓存,保障DNS解析的安全性。

在本研究中,我们分析了3C方法存在一个比较大的性能瓶颈,就是需要经常去做本地缓存解析记录和通过权威查询获得的解析记录的比对。所以为了提升3C方法的效率,同时在启动容灾递归解析系统的时候,我们启动了一个本地顶级权威镜像查询系统提高比对速度,同时提高本地递归解析系统和容灾解析系统的查询效率。尽管本文提出了3C方法来防范DNS缓存攻击,但是随着攻击技术的不断发展,新的攻击手段可能会出现,我们打算在未来通过简化对比逻辑、减少缓存存储和利用其他辅助服务器的方法来进行一步提高3C方法的效率。

参考文献

- [1] MAN K Y, ZHOU X A, QIAN Z Y. DNS cache poisoning attack: Resurrections with side channels[C]//Proceedings of the 2021

ACM SIGSAC Conference on Computer and Communications Security. New York: ACM, 2021: 3400-3414.

- [2] ALLOWAISHEQ E, TANG S Y, WANG Z H, et al. Zombie awakening: Stealthy hijacking of active domains through DNS hosting referral[C]//Proceedings of the 2020 ACM SIGSAC Conference on Computer and Communications Security. New York: ACM, 2020: 1307-1322.
- [3] DUAN H Y, FISCHER R, LOU J, et al. RHINE: Robust and high-performance Internet naming with E2E authenticity[C]//20th USENIX Symposium on Networked Systems Design and Implementation. Boston: USENIX, 2023: 531-553.
- [4] ZHENG X F, LU C Y, PENG J, et al. Poison over troubled forwarders: A cache poisoning attack targeting DNS forwarding device[C]//Proceedings of the 29th USENIX Conference on Security Symposium. New York: ACM, 2020: 577-593.
- [5] 王文通, 胡宁, 刘波, 等. DNS安全防护技术研究综述[J]. 软件学报, 2020, 31(7): 2205-2220.
- [6] MUSASHI Y, KUMAGAI M, KUBOTA S, et al. Detection of kaminsky DNS cache poisoning attack[C]//2011 4th International Conference on Intelligent Networks and Intelligent Systems. Piscataway: IEEE, 2011: 121-124.
- [7] SON S, SHMATIKOV V. The hitchhiker's guide to DNS cache poisoning[C]//Security and Privacy in Communication Networks: 6th International ICST Conference. Berlin: Springer, 2010: 466-483.
- [8] KHORMALI A, PARK J, ALASMARY H, et al. Domain name system security and privacy: A contemporary survey[J]. Computer Networks, 2021, 185: 107699.
- [9] DAVIS J, DECCIO C. A peek into the DNS cookie jar: An analysis of DNS cookie use[C]//International Conference on Passive and Active Network Measurement. Berlin: Springer, 2021: 302-316.
- [10] YU H S, LIU Y, DUAN L H, et al. Cache Top-level domain locally: Make DNS respond quickly in mobile network[C]//2022 IEEE International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom). Piscataway: IEEE, 2022: 1205-1210.
- [11] LIU Y, YU H S, WANG W Y, et al. A robust blockchain-based distribution master for distributing root zone data in DNS[J]. The Computer Journal, 2022, 65(11): 2880-2893.

作者简介

杨天乐 (1981—), 硕士研究生, 高级技术职称, 现任中广移动通信网络有限公司技术专家, 负责IP/传输/无线等网络新技术研究等工作。

谌颖 (1981—), 硕士研究生, 教授级高工, 现任中广移动通信网络有限公司技术专家。

王海涛 (1970—), 本科学历, 高级技术职称, 现任中广移动通信网络有限公司技术专家。