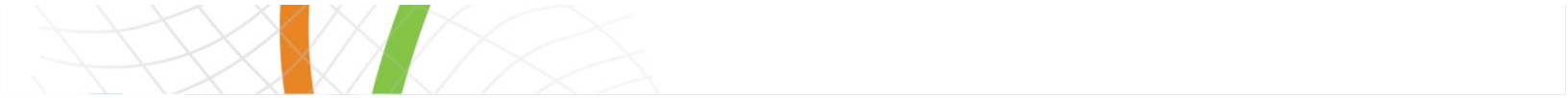
F5分布式拒绝服务保护：推荐实践（第1卷）

如今，从高级金融行业品牌到服务提供商，分布式拒绝服务（DDoS）已成为许多组织的首要任务。经验丰富的管理员知道F5设备不仅非常适合...

白皮书



1 概念

如今，从高级金融行业品牌到服务提供商，分布式拒绝服务（DDoS）已成为许多组织的首要任务。经验丰富的管理员知道F5设备不仅非常适合缓解分布式拒绝服务攻击，有时还是唯一可以缓解某些类型的分布式拒绝服务的设备。许多管理员不知道的是，通过补充F5产品可以实现完整的内部分布式拒绝服务解决方案。

分布式拒绝服务攻击可能是一种压力大的攻击，网络的某些部分将无响应，并且设备可能会全面故障。现在还不是规划防御计划的时候。在“和平时间”期间准备网络应用程序将有助于您缓解未来的攻击。

本指南假定您具有F5网络解决方案和可选的F5安全解决方案。

除非另有说明，否则所有配置，命令和平台均假定为TMOS 11.3.0。

即使许多技术信息特定于F5设备，但某些策略（例如使用SNAT池避免端口耗尽）也可能适用于其他供应商的设备。

2 分布式拒绝服务架构

可以构建耐分布式拒绝服务的应用程序交付网络。本节讨论在攻击之前为使网络和应用程序具有复原力而可以进行的工作。

2.1 F5的推荐架构

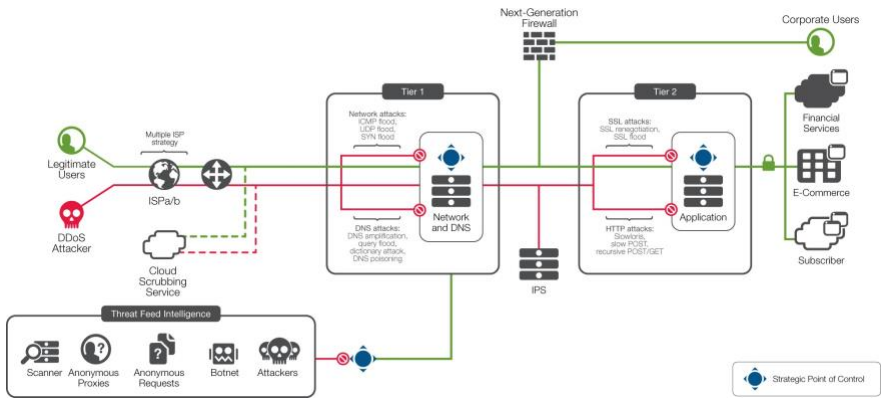


图1：F5建议采用两层分布式拒绝服务方法

许多组织正在重新设计其分布式拒绝服务抵抗架构。对于许多客户，F5建议使用两层分布式拒绝服务解决方案，其中第一层（外围）层由第3层和第4层网络防火墙以及简单的负载平衡组成，而第二层则提供更复杂（且CPU占用更多）的服务，包括SSL终止和网络应用防火墙。

两层方法有多个优点：

- 可以隔离缓解，以便在第1层缓解第3层和第4层，在第2层应用保护。
- 可以相互独立地缩放层。例如，如果网络应用防火墙（WAF）使用量增加，则可以将另一台设备（或刀片）添加到第二层，而不会影响第一层。
- 这些层可以具有不同的平台类型，甚至可以有不同的软件版本。
- 当在第二层应用新策略时，第一层只能将流量的一部分引向新策略，直到对它们进行完全验证。

第1层，第2层DMZ			
F5组件	AFM + LTM	LTM + ASM	GTM DNS Express
OSI模型	第3 + 4层	第7层以上	域名系统
能力	网络防火墙	SSL终止	
	第一—层级负载平衡	网络应用程序防火墙	域名解析
	IP信誉黑名单	平衡	
缓解攻击	SYN洪水	Slowloris	
	ICMP泛洪	慢速发布	UDP泛洪
	格式错误的数据包	阿帕奇杀手	域名系统洪水
	TCP泛洪	鲁迪/保持死	NXDOMAIN洪水
	已知不良演员	SSL重新协商	域名系统

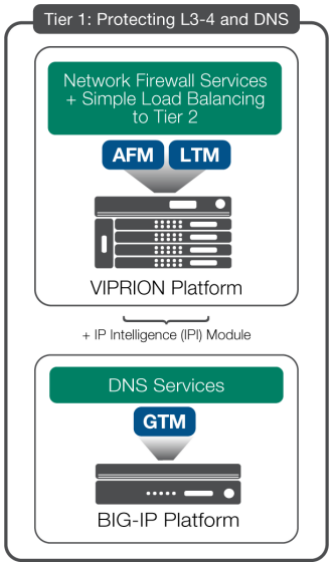
2.2 第1层：网络防御

第一层围绕网络防火墙构建。您几乎可以肯定已经有了一个网络防火墙（可能是或不是F5）和一个网络防火墙团队（或至少一个管理员）。在这一层，您将围绕第3层和第4层（IP和TCP）准备防御措施。在此处，您可以在分布式拒绝服务攻击期间缓解SYN流，TCP流和阻止源地址。

以下各节适用于第1层设备，无论是F5 AFM防火墙模块还是其他供应商网络防火墙前的F5 LTM负载平衡器。

2.2.1 选择虚拟服务器类型

在第1层使用F5防火墙（AFM）或F5负载平衡器（LTM）的组织可以选择如何配置其结构。定义“侦听”对象有四个选项。尽管所有这些都是配置配置的有效方法，但在处理分布式拒绝服务时，有些优势不同。



- **完全代理虚拟服务器是F5配置中的标准虚拟服务器。这些侦听器在与服务器建立辅助连接之前，会与每个传入客户端建立真正的连接。甚至在调用第二层之前，终止和验证客户端连接的行为也提供了广泛的保护。**
- **转发虚拟服务器的运行速度更快，仍然可以抵御SYN洪水，但没有像完整代理虚拟服务器那样提供更广泛的保护级别。**
- **通配符虚拟服务器允许将防火墙规则与应用程序虚拟服务器解耦。这样就可以创建一条规则，说明“对于任何提供FTP服务的地址，请应用此规则集，此镜像策略和此源NAT策略”。**
- **路由域在服务提供商环境中很常见，它将重复的IP子网隔离到逻辑的单独路由表中。尽管路由域本身对分布式拒绝服务没有多大好处，甚至没有，但可以用作挂接第4层安全策略的挂钩。**

图2：通配符服务器是第1层的一个选项

```
ltm virtual ws_ftp {
destination 0.0.0.0:ftp
ip-protocol tcp
profiles { ftp { } tcp { } }
translate-address disabled
}
```

一般而言，当分布式拒绝服务成为首要任务时，F5建议在第1层使用完全代理或转发虚拟服务器。

2.2.2 缓解第1层的SYN洪水

始终通过F5缓解TCP SYN洪水。在11.5版中，F5甚至针对直接服务器返回（DSR）虚拟服务器迁移SYN洪水。为验证您的BIG-IP正在管理SYN缓冲保护，您可以使用简单的show命令查看每个虚拟服务器的SYN缓冲统计信息。

```
% tmsh show ltm virtual vip1
…

SYN Cookies

 Status full-software

 Hardware SYN Cookie Instances 0

 Software SYN Cookie Instances 2

 Current SYN Cache 0

 SYN Cache Overflow 0

 Total Software 432.2K

 Total Software Accepted 0

 Total Software Rejected 0

 Total Hardware 0

 Total Hardware Accepted 0
```

许多F5平台可以缓解硬件中的SYN洪水，这使主要流量控制CPU可以执行其他任务。

平台	硬体	SYN 每 第二	版本号
B4300 Blade	80M	11.3	
B2100 Blade	40M	11.3	
10200V	80M	11.3	
10000S	40M	11.4	
7200V	40M	11.4	
7000S	20M	11.4	
5200V	40M	11.4	
5000S	20M	11.4	
*较旧的平台，包括8800、8400、6800和6400，还支持硬件SYN cookie；但是，版本11.3不支持这些模型，这是本文档的基础。			

表1：SYN Flood硬件支持平台列表

要为特定的虚拟服务器启用用于SYN缓解的洪水硬件，请创建具有更严格安全状态的TCP配置文件。本示例设置两个与分布式拒绝服务相关的变量。它启用硬件SYN cookie。它还设置了递延接受变量，以减少“零窗口” TCP攻击可能对虚拟服务器的影响。

```
% tmsh create ltm profile tcp tcp_ddos { hardware-syn-cookie deferre
d-accept
enabled zero-window-timeout 10000 }
```

然后，通过替换现有的“ tcp”配置文件，将新的tcp配置文件与虚拟服务器关联。

```
% tmsh list ltm virtual vip1 profiles

% tmsh modify ltm virtual vip1 profiles replace-all-with { tcp_ddos
my_ddos1
http }
```

2.2.3 在第1层拒绝UDP和UDP泛洪攻击

UDP洪水是一种常见的分布式拒绝服务矢量，因为它们易于生成且难以防御。通常，除非虚拟主机背后的应用程序正在主动接受该协议，否则不允许向虚拟服务器发送UDP流量。

即使对于接受UDP的应用程序，UDP洪水也可能使系统不堪重负，您可能会发现有必要暂时拒绝到应用程序虚拟服务器的UDP流量。

```
% tmsh create security firewall rule-list drop_udp { rules add { dro
p_udp_rule
{ action drop ip-protocol udp place-after first } } }

% tmsh modify ltm virtual vip1 fw-rules { drop_udp_vip1 { rule-list
drop_udp }
} }
```

攻击停止后，您可以从虚拟服务器中删除规则。

11.5版可以监控和缓解UDP洪水（带细化异常）。这使UDP流量基线能够通过第1层虚拟服务器。如果UDP流量超过阈值，则将其丢弃，除非它与八个用户定义的端口异常（例如，RTSP或DNS）之一匹配。

2.2.4 拒绝ICMP泛洪攻击

ICMP是另一种常见的分布式拒绝服务媒介。 ICMP碎片易于生成和欺骗，并且可以占用许多不同类型的网络设备上的资源。

原子力显微镜可以根据流量模式分析区分正常的ICMP流量和ICMP流量。在虚拟服务器上启用AFM的网络防火墙后，它将监控几种流量的增长。允许使用正常量，其余的则禁止。

Details

#	Attack ID	Attack Type	Virtual Server	Allowed Requests	Dropped Requests	Total Requests
1	129352313	ICMP flood	/Common/wildcard_vs	21,410	293,107	314,517

2.2.5 使用AFM的分布式拒绝服务设备配置文件

攻击者可以用大量的特制无效数据包来消耗防火墙资源。防火墙需要查看（并记录）每个数据包。 F5发现，信号的可疑组合（如带有空有效载荷的PSH + ACK）可能会流行一个月，随后被抛弃，转而采用另一种组合。

这种不断变化的态势很难预测L3 / L4攻击可能会发生什么。安全管理员（对于其他供应商的防火墙）应了解这些攻击，并准备插入规则加以阻止，并避免使用过多的CPU。

F5解决此问题的方法是将大部分L3 / L4协议验证转移到支持它的TMOS平台上的自定义硬件逻辑中。缺省情况下，AFM模块监控数十种第3层和第4层分布式拒绝服务攻击媒介，如圣诞树数据包或陆地攻击数据包等。无论任何BIG-IP设置如何，几乎所有这些数据包都将被丢弃。当检测到这些数据包洪水时，AFM可以发送特殊日志消息。

表1显示了哪些TMOS平台支持硬件辅助的L3 / L4协议验证。这些平台具有SYN洪水硬件支持。

所有平台（包括虚拟版本）都可以管理跟踪这些L3 / L4可疑数据包流的参数。管理屏幕可从用户界面的“安全性”选项卡访问。然后选择“拒绝服务保护和设备配置”。

Attack Type	Detection Threshold PPS	Detection Threshold Percent	Default Internal Rate Limit
L2 Length >> IP Length	10000	500	100000
IPV6 Fragment	10000	500	100000
Payload Length < L2 Length	10000	500	100000
TCP Header Length Too Short (Length < 5)	10000	500	100000
IPV6 Source Address == Destination Address	100	500	1000
FIN Only Set	10000	500	100000
Header Length > L2 Length	10000	500	100000
Bad IPV6 Version	10000	500	100000
Bad IPV6 Hop Count	10000	500	100000
Bad TCP Checksum	10000	500	100000
IPV6 Length > L2 Length	10000	500	100000
ICMP Flood	100	500	500
Bad UDP Checksum	10000	500	100000
IP Length > L2 Length	10000	500	100000
IPV6 Extended Header Frames	10000	500	100000

图3：网络分布式拒绝服务配置设置

这些设置也可以通过命令行与安全dos设备配置命令一起使用。还要注意，这些设置是基于流量管理微内核（tmm）的，而不是基于平台的。在表中，列映射到这些值。

- **检测阈值PPS。这是BIG-IP系统用来确定攻击是否正在发生的每秒（这种攻击类型）数据包数量。当每秒的数据包数量超过阈值数量时，**

BIG-IP系统记录并报告攻击，然后继续检查
每秒，并将超过阈值的阈值标记为攻击。

- **检测阈值百分比。这是表明攻击正在发生的百分比增加值。 BIG-IP系统将当前速率与最近一小时的平均速率进行比较。例如，如果最近一小时的平均速率为每秒1000个数据包，并且将百分比增加阈值设置为100，则检测到的攻击比平均水平高100%，即每秒2000个数据包。超过阈值时，将记录并报告攻击。然后，BIG-IP系统会自动建立一个速率限制，该速率限制等于最近一小时的平均值，所有超过该限制的数据包都将被丢弃。 BIG-IP系统继续每秒检查一次，直到传入数据包速率下降到百分比增长阈值以下。速率限制一直持续到速率再次降至指定限制以下。**
- **默认内部速率限制。这是此类数据包不能超过的值（以数据包每秒为单位）。丢弃所有超过阈值的此类数据包。速率限制一直持续到速率再次降至指定限制以下。**

2.2.6 缓解TCP连接泛洪

TCP连接流是第4层异常，可能会影响网络上的所有有状态设备，尤其是防火墙。通常，这些洪水没有实际内容。第一层的LTM或AFM可以通过将连接吸收到大容量连接表中来缓解这种情况。

平台	TCP连接	桌子 尺寸	SSL连接	桌子 尺寸
VIPRION 4480（4 X B4300）	1.44亿	3200万		
VIPRION 4480（1 X B4300）	3600万	800万		
VIPRION 4400（4 X B4200）	4800万	500万		
VIPRION 4400（1 x B4200）	1200万	100万		
VIPRION 2400（4 x B2100）	4800万	1000万		
VIPRION 2400（1 x B2100）	1200万	250万		
11000系列	24-30百万	264-390万		
10200系列	3600万	700万		
8900系列	1200万	264万		

平台	TCP连接	桌子 尺寸	SSL连接	桌子 尺寸
7000系列	2400万	400万		
6900系列	600万	66万		
5000系列	2400万	400万		
4200V系列	1000万	240万		
3900系列	600万	66万		
虚拟版	300万	66万		

2.2.7 配置自适应收割

即使使用大容量连接表，仍有一些设置可以调整，以加深针对洪水攻击的保护配置文件。

如果BIG-IP连接表变满，将根据自适应收割的低水位和高水位设置“收割”连接。可以从默认值85和95向下调整这些参数，以开始更快地缓解“尖峰”分布式拒绝服务攻击，从而减小初始攻击加载服务器的窗口。

```
% tmsh modify ltm global-settings connection adaptive-reaper-lowater
 75
```

2.2.8 修改空闲超时以应对空连接泛洪

虽然第4层连接流通常不会对F5设备构成高风险，但对其他有状态设备（如其他防火墙）肯定有影响。这些设备几乎总是会在F5状态表填满之前崩溃（请参阅2.2.6节表2）。如果连接流主要由空连接组成，则可以指示BIG-IP更加主动地关闭这些空连接。

与BIG-IP的第4层相关的三个主要配置文件：

- fastL4-硬件辅助的高性能TCP配置文件
- tcp-大多数虚拟服务器使用的标准TCP配置文件
- udp-标准UDP配置文件

注意：您可能会看到其他基于tcp或udp配置文件的内容，如与WAN优化相关的内容。

使用这些配置文件的以下属性控制连接被BIG-IP关闭之前的空闲时间。在严重攻击期间，使用越来越小的值。

对于fastL4配置文件，请覆盖超时重置值和空闲超时值。默认超时为300秒，应在攻击期间进行大幅调整。

```
 % tmsh create ltm profile fastl4 fastl4_ddos { reset-on-timeout dis
abled idle- timeout 15 }
```

对于每一个遭受攻击的fastL4虚拟服务器，用新服务器替换fastL4配置文件。

对于tcp配置文件，出于相同的原因覆盖相同的两个值。当您在那里时，可能还需要调整hardware-syn-cookie和零窗口超时值。参见2.2.2节。

对于udp配置文件，仅降低空闲超时值（默认值为60秒）。

2.2.9 控制速率整形

可以快速部署的另一种防御技术是速率整形。速率整形可以限制BIG-IP上的入口流量速率，并且可能是抵御容量攻击的最简单方法。速率整形虽然功能强大，但防御分布式拒绝服务却不是理想的技术。由于速率调整不能区分好请求和坏请求，因此也可以舍弃您的良好流量，这可能并不是您想要的。

您可以手动配置速率整形配置文件，然后将其分配给虚拟服务器。

在本例中，速率调整类名为“ protect_apache”保证至少有1mbs的流量到达目标，但不允许超过10mbs。

```
net rate-shaping class protect_apache { rate 1mbps ceiling 10mbps }
```

然后，将此速率调整类应用于每个目标虚拟服务器。

2.2.10 设置ICMP最大拒绝速率

TM.MaxRejectRate系统变量可以通过限制BIG-IP系统响应无法与虚拟服务器连接匹配的传入连接发送的TCP RST或ICMP不可达数据包的数量，从而减少拒绝服务攻击的影响。 TM.MaxRejectRate系统变量的默认值为每秒250个TCP RST或250个ICMP不可达数据包。

将该值降低到100可有助于减少出站拥塞，而不会影响网络性能。

```
% tmsh modify sys db tm.maxrejectrate value 100
```

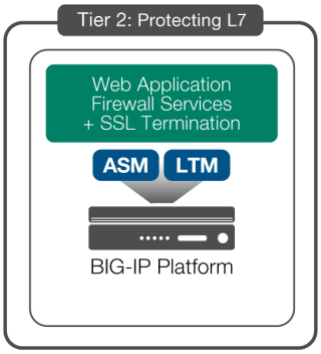
2.3 第2层-应用程序防御

第二层是部署应用程序感知，CPU密集型防御机制，如登录墙，网络应用防火墙策略和LTM iRules。第2层通常也是SSL终止的地方。尽管有些组织在第1层终止SSL，但由于SSL密钥和策略对将其保持在安全范围内的敏感度较高，因此在第1层不那么常见。

2.3.1 了解GET洪水

递归GET和POST属于当今最有害的攻击。他们很难与合法流量区分开。

GET洪水会使数据库和服务器不堪重负。 GET泛洪还可能导致“反向全管道”。 F5记录到一个攻击者向受害者发送了100Mb GET查询，并带出20Gbs数据。



如果您有基于签名的抗分布式拒绝服务解决方案（来自F5或其他供应商），请利用它来保护您的应用程序。借助LTM和ASM，F5提供了许多不同的方法来缓解困难的应用程序层攻击。

GET洪水的缓解策略包括：

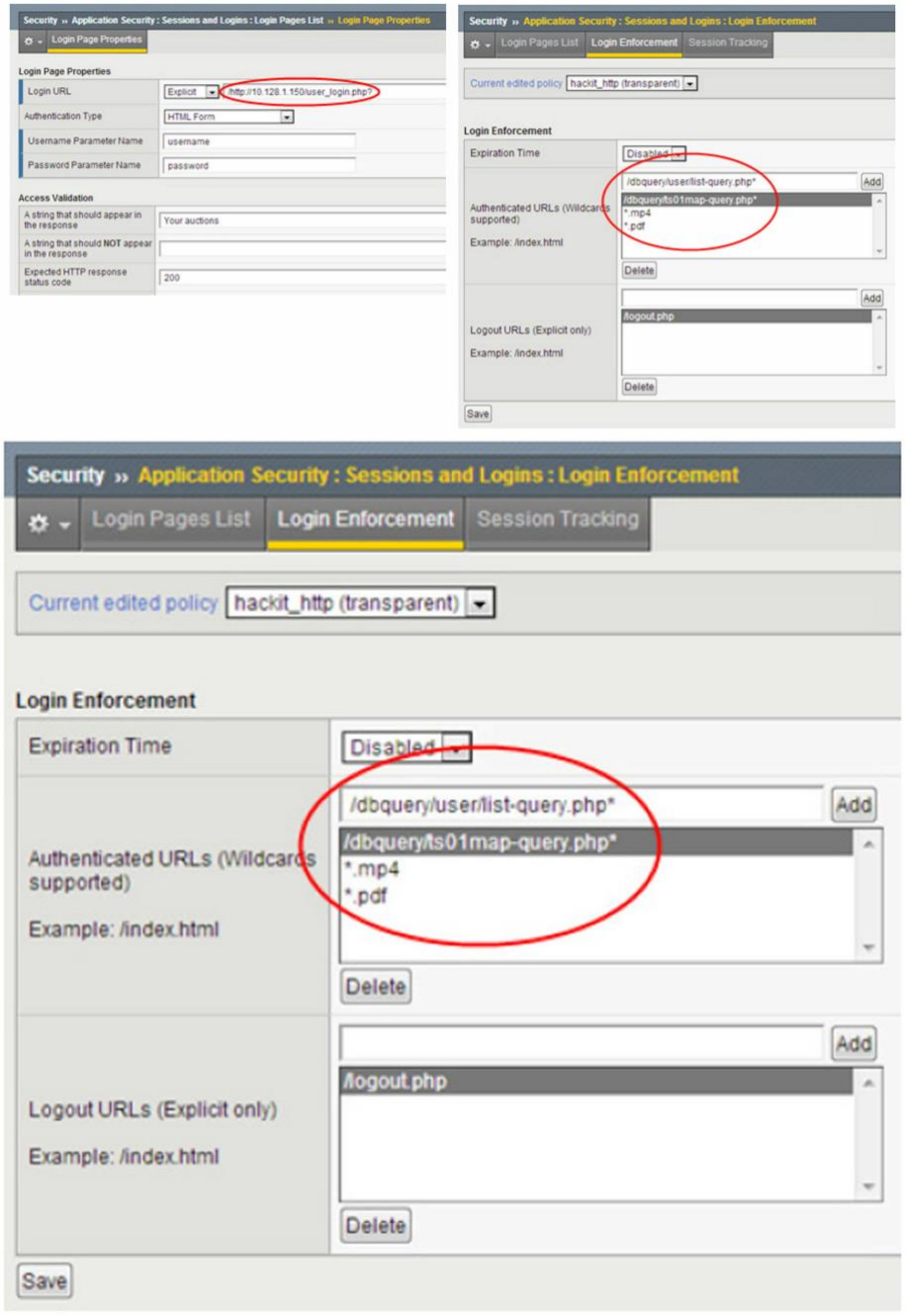
- 登录墙防御
- 分布式拒绝服务保护配置文件
- 真正的浏览器强制执行
- 验证码
- 请求限制iRules
- 自定义iRule

2.3.2 通过配置登录墙减少威胁面

阻止应用程序级攻击的最强大技术是仅允许经过身份验证的用户访问应用程序的数据库部分。创建登录墙可能是一项微妙的工作，在和平时期而不是在繁忙的分布式拒绝服务攻击期间，更好的完成。请注意，并非所有应用程序都可以依赖注册用户并必须处理匿名流量，但对于那些可以使用的应用程序，登录墙是防御之道。

2.3.2.1 使用ASM指定登录墙

ASM通过使用登录页面和登录强制，提供了在ASM策略中执行此操作的功能。此功能将强制用户在一组登录页面上成功进行身份验证之前，不得与一组URL交互。



首先，从安全性→应用程序安全性→会话和登录屏幕定义登录页面。

然后使用“登录实施”选项卡指定需要保护的页面。理想情况下，这些对象将是大型对象，如.MP4和.PDF以及任何可用于非对称攻击的数据库查询。

有关登录强制功能的完整说明，请参阅ASM配置指南中的“创建登录页面”一节。

注意：如果不确定要保护哪些资源，可以“重新定义自己的应用程序”>“重新定义自己的应用程序”（请参阅第3.2.2节）。

2.3.2.2 编写登录墙脚本

您可以通过在登录页面上设置特定的cookie，然后在其他页面上检查该cookie，仅使用LTM iRule创建登录墙。创建此iRule，附加并测试。然后分离它，并将其保存在库中以根据需要激活。

这是DevCentral上的链接。login-wall iRule

2.3.2.3 使用拒绝服务保护文档保护应用程序

F5网络应用程序防火墙ASM包含应用程序特定的“拒绝服务配置文件”。这些强大的配置文件通过监控服务器延迟或http请求速率来检测拒绝服务条件。然后，随着攻击缓解，ASM可以触发可选的iRule事件。

缓解措施包括：

使用以下命令创建拒绝服务配置文件并将其附加到应用程序：

```
% tmsh create security dos profile my_dos_prof { application add { L
rule1 {
latency-based { url-rate-limiting enabled mode blocking } } } }

% tmsh modify ltm virtual my_vip1 profiles add { my_dos_prof }
```

您可以从“安全性”选项卡访问此拒绝服务配置文件。然后选择“拒绝服务保护”。在该屏幕上，检查应用程序安全性，然后配置L7DOS保护参数。

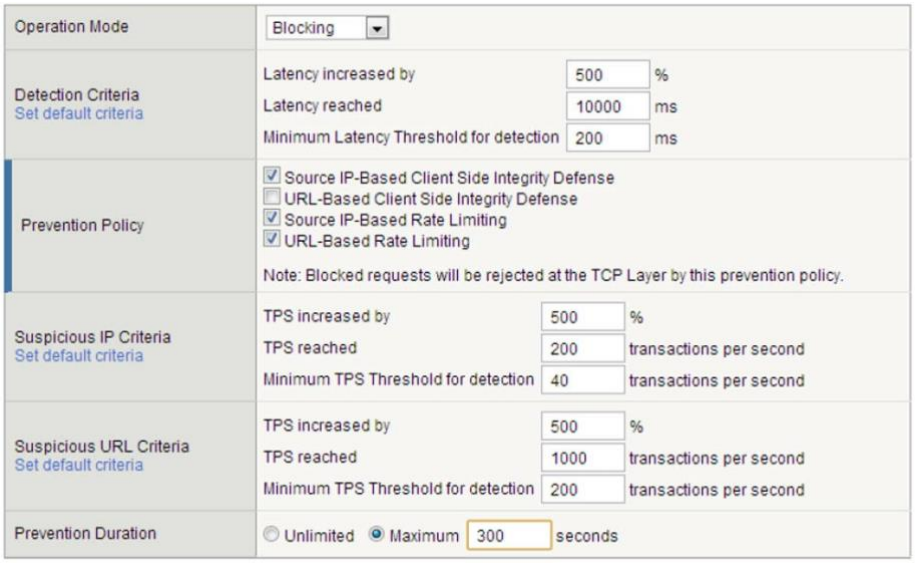


图4. ASM模块的全面L7DOS保护配置

2.3.2.4 实施真实浏览器

除了身份验证和基于tps的检测（第2.3.2.3节）之外，F5设备还可以通过其他方法将实际的网络浏览器与可能的机器人分开。

使用ASM，最简单的方法是创建拒绝服务保护配置文件，然后打开“基于源IP的客户端完整性防御”选项。这将向客户端流注入JavaScript重定向，并在首次看到源IP地址时验证每个连接。

图5.插入JavaScript重定向以验证真实的浏览器

从命令行：

```
 % modify security dos profile my_ddos1 application modify { Lrule1
{ tps-based { ip-client-side-defense enabled } } }
```

2.3.2.5 通过脚本进行的GET请求请求泛洪

F5 DevCentral社区开发了几种强大的iRules，可自动限制GET请求。客户将不断完善这些技术以跟上当前的攻击技术。

这是iRules之一，非常简单，可以在本文档中表示。实时版本可在以下DevCentral页面上找到：HTTP-Request-Throttle

```
when RULE_INIT {

 # Life timer of the subtable object. Defines how long this object e
xist in the
subtable

 set static::maxRate 10

 # This defines how long is the sliding window to count the requests
.

 # This example allows 10 requests in 3 seconds

 set static::windowSecs 3

 set static::timeout 30

}

when HTTP_REQUEST {

 if { [HTTP::method] eq "GET" } {

 set getCount [table key -count -subtable [IP::client_addr]]

 if { $getCount < $static::maxRate } {

 incr getCount 1

 table set -subtable [IP::client_addr] $getCount "ignore"
$static::timeout $static::windowSecs

 } else {

 HTTP::respond 501 content "Request blockedExceeded requests/sec
limit."
```

```
  return
  }
  }
}
```

另一个实际上是上述衍生的iRule是高级版本，还包括一种从iRule本身内部管理被禁IP地址的方法：

- 删除可疑连接。
- 向客户端返回JavaScript重定向，以强制使用浏览器。
- 通过客户端地址或URI进行速率限制。
- URI-Request Limiter iRule-将过多的HTTP请求拖放到特定的URI或从IP

2.3.2.6 使用验证码消除僵尸机器人

缓解GET洪水的另一种方法是使用验证码机制验证“人性”。验证码机制向用户显示加扰单词的图片，用户通过将单词输入网络形式证明其人性。即使黑客和研究人员试图“破坏”计算机，但验证码仍然是区分人与计算机的最佳方法之一。模式识别算法的进步似乎使攻击者接近使验证码系统自动化。但是，根据F5的经验，“破坏”验证码所需的计算工作会大大降低现代分布式拒绝服务攻击者的不对称优势，目前仍保持这些攻击理论。这意味着验证码仍然是拒绝僵尸网络的有效手段。

Google提供了reCAPTCHA服务，该服务执行此功能，同时还能解码远古文本。DevCentral上有一个Google，可用于验证人在连接的另一端。下载iRule（大约150行），并对其进行编辑以提供一些基本信息（例如Google reCAPTCHA密钥和域名系统服务器）。使其在您的BIG-IP上可用。将其连接到虚拟服务器并进行测试，然后为部署做好准备。ReCAPTCHA iRule

2.3.3 编写自定义缓解脚本

如果必须排除所有其他技术，可能会发现有必要编写一个自定义iRule来保护应用程序免受应用程序层攻击。这些自定义规则通常分为两类：过滤和不加区分的阻止。

虽然这可能是本文档中所有技术中最“手动”的技术，但它也是敏捷F5客户中功能最强大和使用最多的技术。 F5 iRules具有极高的可编程性，只要管理员能够编写足够好的脚本，管理员就可以阻止几乎所有类型的攻击。安全相关的iRules如今保护着许多组织，是F5区分应用程序层分布式拒绝服务攻击的真正区别之一。

如果攻击使您无法进行出站互联网访问，请在devcentral.f5.com上搜索一些可能与攻击匹配的关键字。您可能会发现已经为您编写了一个iRule！

要编写自己的规则，首先要剖析攻击流量，并找到与传入攻击流量有关的功能，您可以使用该功能区分正常流量和不良流量。然后编写一个iRule来检测该流量并将其删除。如果您不是iRule的作者，本文档（及整个文档）中散布着iRules，您可以作为示例。将新的iRule附加到应用程序的虚拟服务器。DevCentral

一个简单的安全iRule的例子就是早期的Dirt Jumper iRule，它指出了恶意软件在引荐来源网址中不包含//的事实。

```
when HTTP_REQUEST {
 if { [HTTP::header exists "Referer"] } {
 if { not ([HTTP::header "Referer"] contains "\x2F\x2F") } {
 drop
 }
 }
}
```

如果无法轻松区分好流量和坏流量，可以编写一个iRule，根据请求的对象丢弃流量。例如，如果攻击者正在请求特定的大型PDF或MP4文件，则可以使用iRule删除对该对象的所有请求。

```
 ltm data-group internal block_uris { records { /faqs/faq.mp4 { } /l
ocator/locations.pdf { } /cgi-bin { } } type string }
```

您也可以使用BIG-IP以外的主机。external data groups

然后，使用简单的清理器iRule删除请求URI的请求，该URI与数据类匹配。

```
ltm data-group internal block_uris {
 records {
 /faqs/faq.mp4 { }
 /locator/locations.pdf { }
 /cgi-bin { }
 }
 type string
}
```

绝对不是最好的解决方案，因为它将消除好与坏的流量。它可以使服务器保持活动状态，但是如果您有时间和能力编写上述规则，通常可以找到区分好流量和坏流量的方法。

2.3.4 第2层缓解SSL分布式拒绝服务

尽管可能甚至有时更愿意在任一层终止SSL，但F5建议使用物理（非虚拟）设备在2层终止SSL。通过F5中使用的SSL加速硬件的存在，可以缓解许多SSL分布式拒绝服务攻击。物理设备。其中包括：

- SSL协议攻击
- SSL重播攻击
- SSL连接流
  无论是否使用硬件，F5还将通过自适应收割（参见2.2.7节）和大容量连接表（2.2.6节）减轻SSL连接流量。

  SSL重新协商攻击可以通过两种方式之一缓解。在大多数情况下，您可以在虚拟服务器的SSL客户端配置文件中暂时禁用SSL重新协商功能。但是，非常长寿的连接（如自动柜员机或数据库连接）仍需要具有重新协商的能力。

2.3.5 了解连接多路复用和端口耗尽

通常，请勿在第1层执行连接复用和SNAT等功能。这些功能及相关额外功能（如X-Forwarded-For标头的插入）应在第2层处理。

2.3.5.1 连接多路复用

第7层分布式拒绝服务可以耗尽后端资源，如连接表。应对这种影响的一种方法是通过负载平衡器复用连接。在LTM上，此功能称为OneConnect，可将使用的TCP连接数量减少一个数量级，同时仍保持（或什至提高）每秒总体请求。

在用作分布式拒绝服务防御之前，应对每个应用程序测试OneConnect功能。某些应用程序可能依赖于每个用户单独的连接。

2.3.5.2 港口耗尽

一个SNAT支持每个目标IP大约64,000个并发连接。大量请求可能超过64,000个连接限制，并导致TCP端口耗尽。您可以使用SNAT池来克服此限制。在SNAT池中配置并设置适当的IP地址，以缓解耗尽。

例如，如果虚拟服务器vip1使用简单的自动映射源地址转换，则可以使用以下命令将其更改为使用IP地址池。本示例仅使用三个地址将可用端口从64,000增加到192,000。

```
% tmsh create ltm snatpool ddos_snatpool members add { 10.1.20.
161 10.1.20.162
```

```
 % tmsh modify ltm snat-translation 10.128.20.161 { ip-idle-tim
eout 60 }
```

```
l ddos_snatpool }
```

对于添加到SNAT池的每个地址，您可能需要分配一个离散超时值（默认为不确定）。借助空闲超时，BIG-IP可以关闭空闲连接并帮助保护上游有状态防火墙。

对snatpool中的每个地址重复上述命令（在上例中为10.1.20.162和10.1.20.163）。

3 更多分布式拒绝服务推荐实践

3.1 缓解域名解析服务

域名系统是仅次于HTTP的第二大攻击目标服务。当域名系统（DNS）中断时，所有外部数据中心服务（不仅仅是一个应用程序）都会受到影响。单点完全故障，再加上历史上配置不足的域名系统（DNS）基础设施，使得域名系统（DNS）成为攻击者非常诱人的目标。即使攻击者并非专门针对域名系统，也常常会无意间这样做：如果攻击客户端在发起洪水之前都在查询目标主机的IP，结果将是对域名系统的间接攻击。

由于相对简单的基于UDP的域名系统协议，域名系统攻击具有两个主要特征：

- 域名系统（DNS）攻击很容易产生。
- 域名系统（DNS）攻击很难防御。

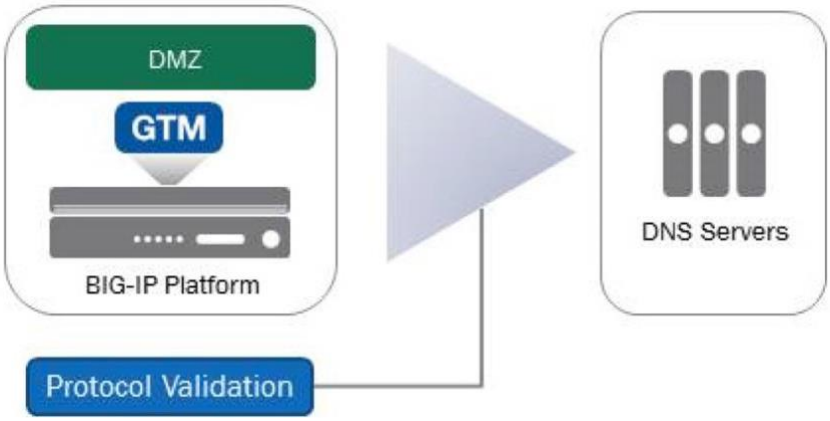


图6：缓解DNS分布式拒绝服务

有四种用于缓解域名系统分布式拒绝服务攻击的内部部署策略：

- 使用协议验证。
- 检测并防止域名系统洪水。
- 针对NXDOMAIN查询流超额配置域名服务。
- 黑名单是不得已的方法。

3.1.1 考虑域名服务的位置

您可能会注意到，在图1中，DNS服务作为其自身的一组设备存在于安全边界之后。通常，在安全层之间使用此所谓的DMZ为DNS提供服务。这样做是为了保持域名系统（DNS）与其服务的应用程序无关-例如，如果数据中心的那部分耗尽，域名系统（DNS）可以将请求重定向到辅助数据中心（或云）。 F5建议采用这种策略，将域名系统与安全和应用程序层分开，以实现最大的灵活性和可用性。

一些拥有多个数据中心的大型企业将进一步发展，并结合使用F5的GTM域名系统快速域名系统和AFM防火墙模块为主要安全边界之外的域名系统服务。这种方法的主要好处是，即使第1层防火墙由于分布式拒绝服务而无法使用，域名系统服务仍然可用。

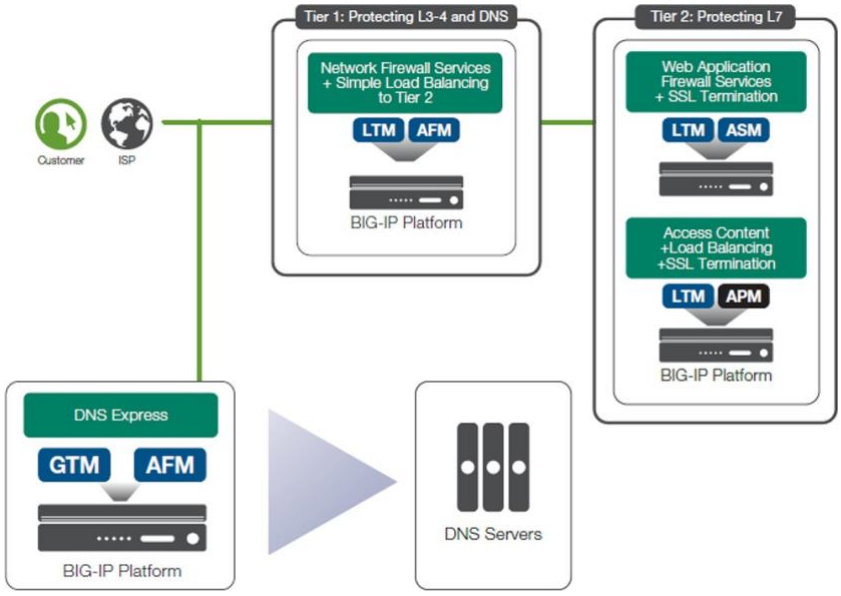


图7：备用外部域名系统架构

3.1.2 使用协议验证保护域名系统服务

无论您在DMZ内部还是外部为域名系统提供服务，都可以使用GTM或AFM在域名系统请求到达域名服务器之前验证域名系统（DNS）请求。

如果您让GTM执行全局服务器负载平衡，则可能已经阻止了许多域名系统拒绝服务攻击。您可以从GTM GUI的主信息中心查看域名系统（DNS）查询/响应性能。由于GTM是域名系统的全代理服务器，因此它将自动验证每个请求并丢弃无效请求。

但是，您可能会发现，看起来有效的请求仍然使服务器不堪重负。如果使用F5防火墙模块AFM，则可以使用协议安全配置文件进一步过滤特定类型的域名系统（DNS）请求。

在“安全性”选项卡中，选择“协议安全”，然后选择“安全配置文件”。选择域名系统，然后按创建按钮。在此屏幕上，您可以构建协议安全配置文件来过滤或阻止不同类型的请求。

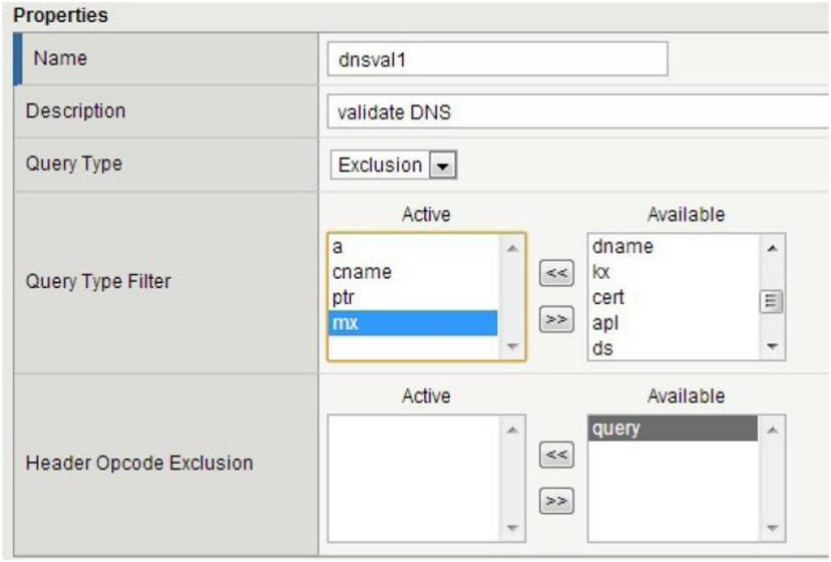


图8：域名系统协议验证

3.1.3 检测域名系统洪水

F5防火墙模块AFM具有强大的DNS分布式拒绝服务功能-可以按记录类型检测DNS洪水。在“安全性”选项卡中，选择“拒绝服务保护”，然后选择拒绝服务配置文件并最终创建。在创建屏幕中，单击域名系统复选框，然后设置和接受阈值参数。

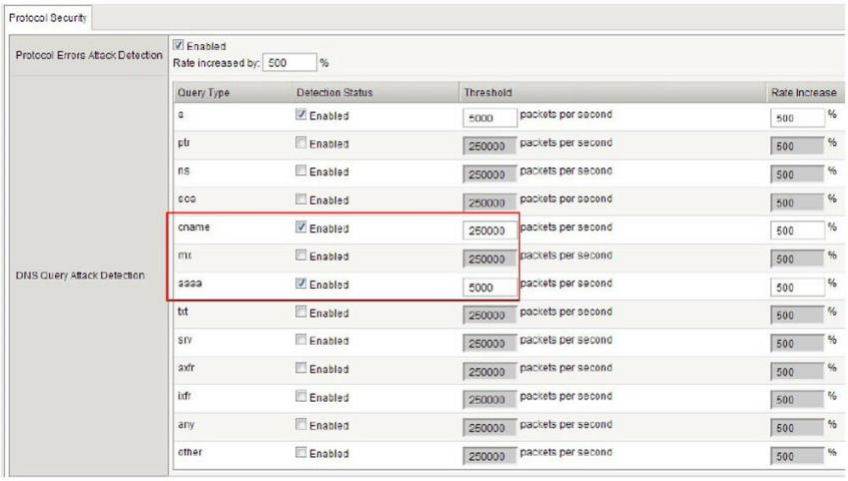


图9：域名系统洪水检测

协议错误复选框表示系统检测到恶意或格式错误的域名系统（DNS）查询，并以百分比显示在系统跟踪格式错误和恶意域名系统（DNS）查询之前，合法的域名系统（DNS）查询流量增长量。

注意：目前，此防火墙功能可检测流量，但不会丢弃数据包以缓解流量。

3.1.4 针对查询泛洪超额配置域名系统服务

域名系统（DNS）服务历来配置不足。造成这种情况的部分原因是，对于许多组织而言，对于任何特定团队而言，域名系统所有权并不是一个积极的发展。无论出于什么真正原因，域名系统部署中有很大一部分配置不足，甚至无法抵御中小型分布式拒绝服务攻击。

域名系统缓存已经可以提高域名系统缓存的感知性能，并且提供了一定的抵御标准域名系统查询攻击的能力，因此变得越来越受欢迎。攻击者已转换为“无此类域”（或NXDOMAIN）攻击，可迅速耗尽缓存提供的性能收益。

F5推荐的补救方法是使用称为DNS Express的特殊高性能DNS代理模块引导DNS服务。 DNS Express充当现有域名服务器之前的绝对解析器。它将从服务器加载区域信息，然后解析每个请求或返回NXDOMAIN。它不是缓存，无法通过NXDOMAIN查询流清空。

在GTM或域名系统服务中，域名系统快速通道每CPU每秒可处理250,000个请求，因此可抵抗除最猛烈的域名系统攻击之外的所有攻击。域名系统（DNS）服务器保持不变，可以管理区域数据。

3.1.5 黑名单作为最后的手段

域名系统流量传统上是UDP，易于生成且容易欺骗。常规的第3层和第4层防御（如按源IP列入黑名单）通常对域名系统（DNS）流无效。实际上，按源IP阻止域名系统（DNS）请求可能非常危险。例如，如果您在不知情的情况下阻止了来自主要互联网服务提供商的请求，则可能拒绝向许多合法用户提供服务而没有意识到。

请参阅“ BIG-IP系统：DOS保护和协议防火墙实施”（第3章－检测和防止域名系统拒绝服务攻击）。

3.1.6 当心分布式拒绝服务确认参与

3.1.6.1 未使用的查询类型

攻击者可以通过发送未使用服务的查询来诱骗域名系统（DNS）服务炸毁第三方目标。使用AFM屏幕（请参见上面的图9）禁用未使用的查询类型。然后，当查询这些类型时，将删除它们。将不提供任何响应，从而帮助避免参与分布式拒绝服务攻击。

对于MX（邮件服务）和区域传输尤其如此。如果您的组织在已知的特定时间进行一次传输，请在其他所有时间禁用IXFR，AXFR和ZXFR类型。

3.1.6.2 域名系统

DNSSEC是全球域名服务的重要发展。最终，它将减少网络钓鱼等欺诈行为。对于域名系统分布式拒绝服务，情况更为复杂。域名系统（SECSEC）响应有时比传统域名系统（UDP）UDP响应大10-20倍。这就是说，DNSSEC服务器实际上是在无意中轰炸无效响应，诱使他们攻击其他计算机。

借助GTM，F5拥有市场上性能最高的域名系统（DNSSEC）解决方案。如果将GTM用作攻击媒介，则可以制造强大的武器。因此，GTM允许您对响应数进行速率限制，以防止自身参与攻击。

```
% tmsh modify sys db dnssec.maxnsec3persec value 10
```

dnssec.maxnsec3persec变量控制GTM每秒发送的NSEC3权威NXDOMAIN

```
% tmsh modify sys db dnssec. signaturecachensec3 value true
```

消息的上限。 0为无限制，默认为0。更高的限制值，例如每秒10至100之间，可能会阻止GTM本身在攻击期间使用。

将dnssec.signaturecachensec3变量设置为false可以完全阻止NXDOMAIN消息使用GTM缓存，从而防止攻击者使用“无此类域”响应填充GTM缓存。

3.2 其他分布式拒绝服务最佳实践准备程序

准备进行分布式拒绝服务攻击的时间将提高防御效率。您可以通过以下几种方法使组织为分布式拒绝服务攻击做好准备。

3.2.1 配置和验证日志

在攻击期间，您很有可能会发送诊断信息并记录异常和流量峰值。应对大型分布式拒绝服务攻击时，高性能至关重要。检测也同样重要，这意味着您将需要使用BIG-IP的高速日志记录功能将此信息发送到第三方记录设备（如Splunk）或SIEM（如ArcSight）。

注意：缓解分布式拒绝服务攻击时，必须使用第1层的BIG-IP高速记录功能。不要使用本地日志记录；强烈的分布式拒绝服务攻击可能会淹没本地基于磁盘的日志记录。

3.2.1.1 设置高速日志记录

- 创建一个池以映射到您的外部日志服务器（在本例中为系统日志）。根据需要重写ArcSight，TrustWave或环境支持的任何SIEM解决方案。然后创建日志配置对象，以正确格式化和转发字符串。

```
% tmsh create ltm pool hsl_pool members add { 10.128.10.25
0:514 }

% tmsh create sys log-config destination remote-high-speed
-log log_dest_HSL {
pool-name hsl_pool }

% tmsh create sys log-config destination remote-syslog log
_dest_format { format
rfc5424 remote-high-speed-log log_dest_HSL }

% tmsh create sys log-config publisher log_pub_ddos { dest
inations { log_
dest_HSL log_dest_format } }
```

- 使用GUI创建一个日志文件。

进入安全>事件日志>日志文件页面。使用以下命令创建日志文件：

配置文件名称	ddos_log_profil
网络防火墙	已启用
网络防火墙：发布者	log_pub_ddos
日志规则匹配	接受，删除和拒绝
日志IP错误	已启用
记录TCP错误	已启用
记录TCP事件	已启用
存储格式	字段列表选择所有可用项目并将其移动到“选定项目”列表中

- 将该日志文件对象与保护应用程序的虚拟服务器相关联。

```
 % tmsh modify /ltm virtual vip1 { security-log-profiles a
dd { ddos_ log_profile } }
```

3.2.2 审查自己的应用程序

现代分布式拒绝服务攻击者会在发起分布式拒绝服务攻击前几天或几周重新审核应用程序。他们将搜寻您的网站并检索每个有效URI的加载时间和数据大小。通过对结果数据集进行排序，它们将快速隔离您最消耗CPU或数据库的查询和最大的对象（如PDF和MP4）。在分布式拒绝服务攻击期间，他们将反复查询这些对象，使基础设施不堪重负。

尽管第3.8节会帮助您缓解此类攻击，但可以通过重新确认自己的应用程序来帮助自己。这样，您可以更深入地了解哪些URI和子系统可能成为目标，并在以后做出更明智的分类决策。

理想情况下，您可以使用LoadRunner等工具或其他性能监控工具为您提供所需的指标。如果缺乏此功能，则检索基本URL，加载时间和数据大小的最简单方法可能是运行wget实用程序，大多数Linux发行版均提供该实用程序。使用以下语法运行它：

```
% wget -r --spider http://10.128.10.150 2>&1 | grep saved

2013-08-25 15:44:29 (2.48 MB/s) - `10.128.1.150/index.html
' saved [22304]

2013-08-25 15:44:29 (5.53 MB/s) - `10.128.1.150/index.php'
 saved [22304]

2013-08-25 15:44:29 (7.06 MB/s) - `10.128.1.150/sell.php'
saved [41695]
```

最后一个数字（方括号中）是请求的数据大小。您必须通过相互减去时间（秒字段）来获得加载时间。

3.2.3 使用iHealth验证现有BIG-IP设备的运行状况

F5提供名为iHealth的基于云的诊断和启发式服务。 iHealth将检查F5设备的配置，并提出建议以保持BIG-IP快速，安全和可用。虽然大多数设置可能更适用于前两个设置，但其中一些设置可以适用于可用性，以及扩展到分布式拒绝服务弹性。

在此示例中，iHealth显示一个SNAT池已配置，没有超时值。这可能会提醒那些注重资源的管理员确保使用的SNAT池
因为经过加固的虚拟服务器应包括空闲超时，以保持连接数量减少并防止上游防火墙倾斜。

有关iHealth的更多信息，请参见。iHealth web site

3.2.4 准备分布式拒绝服务手册

分布式拒绝服务手册或运行手册是一本程序手册，可帮助您的IT员工抵御分布式拒绝服务攻击。一部优秀的游戏手册将帮助新（和现有）管理员抵御分布式拒绝服务攻击。手册应与最新的白名单和联系信息保持最新。

一些组织将针对自己进行定期分布式拒绝服务演练（甚至测试），以保持最新状态并测试剧本。当关键人物不在场时，尝试让员工练习剧本中的程序-攻击并非总是在最方便的时间表上发生。

如果您没有剧本，可以从F5获得。

3.2.5 回顾两层体系结构中的防御策略

前几节中描述的某些防御策略值得回顾，特别是对于使用非F5网络防火墙的管理员而言。

请记住：

- SNAT池可缓解第1层端口的耗尽。
- 第1层的形状流量
- 积极获得TCP连接。
- 仅将黑名单域名系统（DNS）列入黑名单。
- 在第2层实施登录墙和验证码。
- 在第2层禁用可选的占用大量CPU的功能。
- 始终使用高速远程日志记录。

通过实施本最佳实践指南中的建议，您将可以做很多工作，准备针对分布式拒绝服务攻击的应用程序。

4 结论

了解拒绝服务攻击的现代威胁谱非常重要。了解如何利用已有的防御设备就显得尤为重要。

根据资源和需求，您可能已经针对分布式拒绝服务弹性（DDoS Resiliency）重新设计了网络。如果您正在考虑采用这种方法，请密切注意第2.1节中介绍的建议多层体系结构。即使您的网络安全不是完全由F5技术构建的，为分布式拒绝服务目的独立处理第4层和第7层仍然有意义。

通过遵循推荐的实践，您将为网络，应用程序和人员做好防御攻击的准备。

F5推荐的分布式拒绝服务防护实践的最后一步是准备一个分布式拒绝服务手册。这样的剧本是缓解包括工作表和日志在内的攻击的实时程序指南。 F5可以为您提供入门模板。

专家预测，分布式拒绝服务将在很长一段时间内成为互联网上的问题。很快，做好准备不仅是一种选择，而且是一项要求。

附录

应用分类方法与对策

下一节建议针对特定第7层攻击媒介的缓解措施。其中许多是“慢速”攻击，可能特别有害。

内容

- Slowloris
- 保持死
- 低轨道离子炮（LOIC）
- 慢速POST
- 零窗口攻击
- 慢速读取攻击
- 鲁迪
- 阿帕奇杀手
- SSL重新协商
- 污垢跳线iRule

Slowloris

Slowloris是一种常见的HTTP向量，攻击者会（非常缓慢地）发送小HTTP标头，使HTTP会话保持活动状态（例如，每299秒“ X-a：b”）。如果虚拟服务器当前在第4层进行负载平衡，请考虑将其切换到第7层。这将在添加HTTP配置文件时增加一些本机保护。

```
% tmsh list ltm virtual vip1
ltm virtual vip1  {
destination 10.128.10.141:http
profiles {
```

```
}
}

% tmsh modify ltm virtual vip1 profiles replace-all-with {
 tcp http }
```

这将导致BIG-IP吸收Slowloris连接。如果您担心会堆积过多并导致其他设备（如防火墙）出现问题，请使用以下Slowloris iRule删除10秒后未完成的所有连接（可随意调整此数字）。

```
#Slowloris iRule

当CLIENT_ACCEPTED {

 设置hsl [HSL :: open-proto UDP-pool hsl_pool]

 设置rtimer 0

 10000 {后

 如果{不是$ rtimer} {

 删除

 HSL :: send $ hsl“ [IP :: client_addr]删除-连接到o
慢”

 }

 }

}

当HTTP_REQUEST {

 设置rtimer 1

}
```

保持死

此类攻击基于消耗的CPU和RAM。通过使用Keep-Alive和HTTP HEAD方法，可以创建请求流，而无需触发基于服务器打开连接数的防火墙防御。

ASM模块可以禁止HEAD请求（浏览器通常不使用）。您可以通过配置相关应用程序安全策略中的“允许的方法”拒绝HEAD请求。

参见以获取更多信息。solution 12312

低轨道离子炮（LOIC）

低轨道离子加农炮是一个与僵尸网络组织匿名紧密联系的自愿僵尸网络工具。虽然该工具使用SYN流和UDP流，但最著名的是第7层HTTP流。假设已经缓解了SYN和UDP流（请参见2.2.2和2.2.3节），最后一步是缓解LOIC GET流。

通常，最快的方法是根据每个LOIC HTTP请求所包含的攻击“抗议消息”对其进行过滤。使用Wireshark或tcpdump或其他工具隔离消息，然后将该消息添加到数据组。使用%20表示空格。该消息可能会随着时间变化，您可能需要在攻击持续时间内对其进行监控。

请注意，您可以使用在BIG-IP外部托管的外部数据类-请参阅tmsh命令

```
 ltm data-group anonmsgs { records { Somos%20legi { } U%20
dun%20goofed { } } type string }
```

外壳中的“帮助搜索数据组”。

然后，使用简单的清理器iRule删除包含该数据类中任何有效负载的请求。

```
ltm rule loic_defense_rule {

 when CLIENT_ACCEPTED {

 set hsl [HSL::open -proto UDP -pool hsl_pool]

 }

 when HTTP_REQUEST {

 if { [class match [HTTP::uri] contains anonmsgs] } {

 drop

 HSL::send $hsl "Dropped [IP::client_addr] – suspected Low
 Orbit Ion Cannon"

 }

 }

}
```

慢速POST

Slow-POST攻击的重点在于发送具有给定“内容长度”（通常为大数）的POST请求，然后非常缓慢地将消息正文发送到服务器，同时保持较长的空闲时间。服务器继续接收数据时，服务器将保持连接打开状态。如果对服务器执行了大量此类请求，则可能会耗尽连接表，这将导致服务器无法响应其他请求。

如果您拥有ASM模块，则可以使用ASM系统变量屏幕中的两个变量来缓解发布缓慢的问题-导航至安全：选项：应用程序安全：高级配置**：系统变量-并修改以下变量。**

slow_transaction_timeout（默认为10秒）。根据需要降低此值。

max_slow_transactions（默认为25个事务）。根据需要将此值降低到5或更小。

如果没有ASM，请参阅此LTM iRule以缓解Slow-POST。可以在下一节中与慢速读取iRule一起使用（只需将它们作为两个独立的iRules连接），因为慢速读取iRule基于服务器，而慢速发布iRule基于客户端。

零窗口Aack

零窗口攻击是很难检测到的第4层攻击。它的工作原理是建立与目标的TCP连接，请求一些数据，然后将TCP窗口大小设置为零。这使服务器，缓存或中间件的连接停滞。

如果攻击者针对BIG-IP设置了TCP零窗口长度，则可以使用本节中提到的零窗口超时tcp配置文件值
2.2.2缓解。

慢速读取确认

慢速读取攻击的方法是发送合法的HTTP请求，然后从缓冲区读取HTTP响应，速度非常慢，旨在使受害者尽可能多地保持活动状态。

在版本11.3.0中，ASM模块的低速阻止功能适用于入站请求（如慢速POST）。对于慢速读取，请使用以下LTM iRule缓解措施：

```
when SERVER_CONNECTED {

 TCP::collect
```

```
}

when SERVER_DATA {

 set rtimer 0

# Time in milliseconds before HTTP response read is consid
ered slow:

 after 5000 {

 if { not $rtimer} {

 set hsl [HSL::open -proto UDP -pool hsl_pool]

# Slow read detected for this server response. Increment t
he count by adding a
table entry:

# Add the client source IP::port to the subtable with a ti
meout

 table set -subtable "MyApp" "[IP::client_addr]:[TCP::clie
nt_port]"
"ignored" 180

# If we are over the concurrency limit then reject

 if { [table keys -subtable "MyApp" -count] > 5} {

 clientside {reject}

 table delete -subtable "MyApp" "[IP::client_addr]:[TCP::c
lient_
port]"

 HSL::send $hsl "Dropped [IP::client_addr] - reading too s
low"

 }

 }
```

```
 }

 TCP ::通知响应TCP ::

 发布TCP ::收集

}

当USER_RESPONSE {

 设置rtimer 1

}

当CLIENT_CLOSED {

 表删除子表“ MyApp”“ [IP :: client_addr]：[TCP :: c
lient_port]”

}
```

鲁迪

R-U-Dead-Yet（简称“ RUDY”）使用慢速POST和通过长格式字段提交的通用HTTP DoS攻击。

阿帕奇杀手

Apache Killer也称为远程攻击。当客户端浏览器（例如手机浏览器）仅需要文档的一部分时，可以使用HTTP范围标头请求数据的“范围”。如果客户端只需要前100个字节，可以说：

```
Range:bytes=0-100
```

```
Range:bytes=0-,5-1,5-2,5-3,…
```

Apache Killer攻击的工作原理是请求多个重叠范围，使Apache之类的网络服务器感到困惑：

有三种缓解Apache Killer的方法。您可以修改HTTP配置文件，仅删除Range标头。例如，如果您的http文件名为“ http_ddos2”，则可以运行以下命令：

```
 % tmsh modify ltm profile http http_ddos2 { header-erase
range }
```

下面的iRule是一种更外科手术缓解Apache Killer的方法，仅当请求的射程超过五个时，才删除射程请求。

```
when CLIENT_ACCEPTED {

 pcre:"/Range:[\t ]*bytes=(([0-9\- ])+,){5,}/Hi";
}

when HTTP_REQUEST {

 # remove Range requests for CVE-2011-3192 if more than fi
ve ranges are
requested

 if { [HTTP::header "Range"] matches_regex {bytes=(([0-9\-
 ])+,){5,}} } {

 HTTP::header remove Range

 HSL::send $hsl "Client [IP::client_addr] sent more than 5
 ranges. Erasing
range header."

 }

}
```

第三种使用BIG-IP解决方案的缓解方法是，使用以下ASM攻击签名检测使用此技术的攻击并采取相应措施。

SSL重新协商

如果看到特定的SSL客户端进行大量重新协商，则可能遭受了SSL重新协商攻击。缓解此漏洞的最简单方法是禁用虚拟服务器关联客户端文件中的SSL重新协商。但是，如果在缓解攻击的同时仍支持合法客户端的重新协商（如旧的“逐步升级”或服务器级加密浏览器），则可以使用此iRule或其他类似产品。此规则将在一分钟内关闭尝试超过五个重新协商的所有连接：

```
when RULE_INIT {

 set static::maxquery 5

 set static::mseconds 60000

}

when CLIENT_ACCEPTED {

 set ssl_hs_reqs 0

 set hsl [HSL::open -proto UDP -pool hsl_pool]

}

when CLIENTSSL_HANDSHAKE {

 incr ssl_hs_reqs

 after $static::mseconds { if {$ssl_hs_reqs > 0} {incr ssl
_hs_reqs -1} }

 if { $ssl_hs_reqs > $static::maxquery } {

 after 5000

 drop

 HSL::send $hsl "Dropped [IP::client_addr] - too many SSL
renegotiations"

 }
```

```
}
```

污垢跳线iRule

某些版本的“污垢跳线”工具在引荐来源网址字段中不包含//。这是一个简单的iRule，用于检测和丢弃污垢跳线连接。

```
when CLIENT_ACCEPTED {

 set hsl [HSL::open -proto UDP -pool hsl_pool]

}

when HTTP_REQUEST {

 if { [HTTP::header exists "Referer"] } {

 if { not ([HTTP::header "Referer"] contains "\x2F\x2F") }
 {

 HSL::send $hsl "DDoS Dirt-Jumper HTTP Header Structure mi
ssing x2f x2f
Referer protocol identifier from [IP::client_addr]"

 drop

 }

 }

}
```

Download Volume 2

F5 Networks，Inc.
西华盛顿州西雅图，401 Elliott Avenue West，98119 888-882-4447 f5.com

美洲 info@f5.com
亚太地区 apacinfo@f5.com
欧洲/中东/非洲 emeainfo@f5.com
日本 f5j-info@f5.com