10.3.2 意外重定向

在这个情景中，我们又遇到一位不能在工作站上网的用户。然而，不像之前那个用户，她可以访问 Internet，只是不能访问 Google 主页。每次她想访问 Google 的网站时，都被重定向到一个浏览器页面「该页无法显示」。这个问题只影响她一个人。

与之前的情景一样，这是一个只有一些简单交换机和一个简单路由器网关的小型网络。

1．侦听线路

我们一边监听流量，一边让用户尝试浏览 Google 主页，得到 nowebaccess2.pcap 文件。

2．分析

如图 10-17 所示，捕获记录文件以一个 ARP 请求和响应开始。在数据包 1 中，用户计算机的 MAC 地址是 00:25:b3:bf:91:ee，IP 地址是 172.16.0.8，它向网段上的所有计算机发送一个 ARP 广播数据包，试图获得主机 172.16.0.102 的 MAC 地址。我们目前还不认识这个地址。

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	00:25:b3:bf:91:ee	ff:ff:ff:ff:ff:ff	ARP	42	Who has 172.16.0.102? Tell 172.16.0.8
2	0.000334	00:21:70:c0:56:f0	00:25:b3:bf:91:ee	ARP	60	172.16.0.102 is at 00:21:70:c0:56:f0

图 10-17　对网络上另一个设备的 ARP 请求和响应

在数据包 2 中，用户的计算机了解到 IP 地址 172.16.0.102 的 MAC 地址是 00:21:70:c0:56:f0。根据之前的情形，我们猜测这是网关路由器的地址，通过这个地址数据包可以被再次转发到外部 DNS 服务器。然而，如图 10-18 所示，下一个数据包并不是 DNS 请求，而是从 172.16.0.8 到 172.16.0.102 的 TCP 数据包。它设置了 SYN 标志，表明这是两台主机间建立 TCP 连接时握手的第一个数据包 ❶。

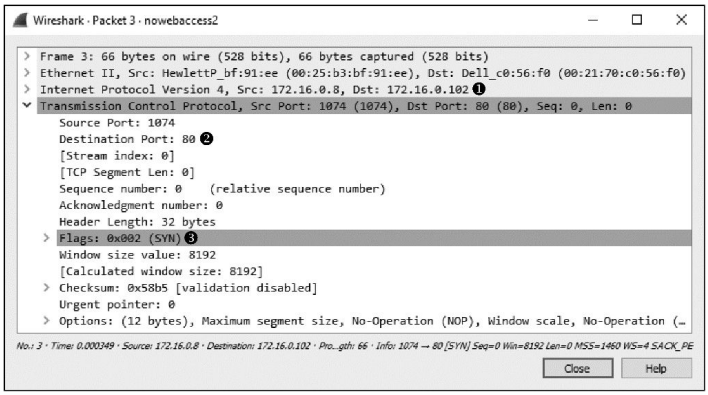

图 10-18　从一台内网主机发往另一台内网主机的 TCP SYN 数据包

显然，试图连接到 172.16.0.102❸ 的 80 端口 ❷ 的 TCP 连接通常与 HTTP 流量有关。如图 10-19 所示，当主机 172.16.0.102 发送回带有 RST 和 ACK 标志 ❶ 的 TCP 数据包（数据包 4）时，连接请求就中断了。

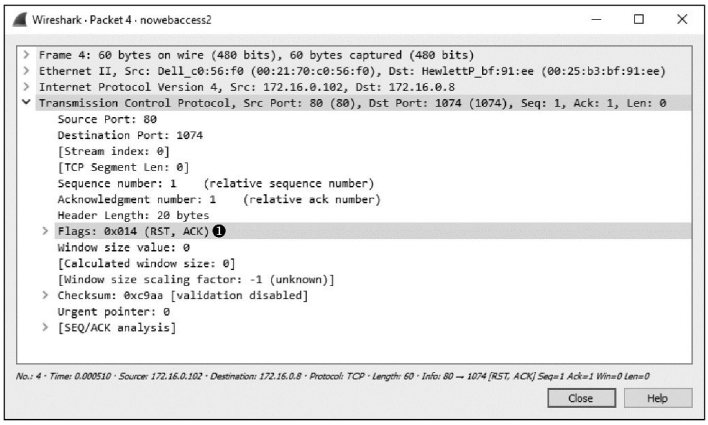

图 10-19　响应 TCP SYN 的 TCP RST 数据包

第 6 章介绍过，带有 RST 标志的数据包是用来结束 TCP 连接的。在这个场景中，主机 172.16.0.8 尝试与主机 172.16.0.102 的 80 端口建立 TCP 连接。不幸的是，由于那台主机没有配置好服务在 80 端口的监听请求，因此只能发送 TCP RST 数据包结束连接。这个过程又重复了两次。如图 10-20 所示，在通信最终结束前，用户计算机发送了一个 SYN 数据包并得到 RST 响应。

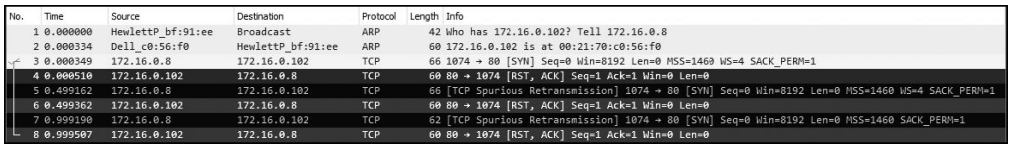

图 10-20　TCP SYN 和 RST 数据包一共出现了 3 次

此时，用户在浏览器上看到了「该页无法显示」。

在查看其他工作正常的网络设备的配置信息后，数据包 1 和 2 中的 ARP 请求和响应引起了我们的注意。因为 ARP 请求并不是指向网关路由器的真

实 MAC 地址，而是其他未知设备。在 ARP 请求和响应之后，我们期望看到向 DNS 服务器的请求，以得到 Google 的 IP 地址，但最终并没有看到。阻止 DNS 查询的两个条件如下。

- 发起连接的设备在 DNS 缓存中已经有域名—IP 地址的对应项。
- 发起连接的设备在 hosts 文件中已经有域名—IP 地址的对应项。

进一步检查这台计算机后，我们发现它的 hosts 文件有一个 Google 表项，对应一个内网 IP 地址 172.16.0.102。这个错误表项就是用户问题的根源。

计算机通常都把 hosts 文件当作域名—IP 地址配对的可信来源，并且会在查询外部来源之前检索它。在这个场景中，用户计算机检查它的 hosts 文件，发现有一个 Google 的表项，就认为 Google 在这个本地网段。接着，它向这个主机发送 ARP 请求，并得到响应，然后尝试向 172.16.0.102 的 80 端口发起 TCP 连接。然而，由于该系统并没有配置成 Web 服务器，因此它不可能接受这个连接请求。

将这个 hosts 文件的表项移除后，用户的计算机就能正常访问 Google 了。

> 注意
>
> 在 Windows 系统上查看 hosts 文件，请打开 C:\Windows\System32\drivers\hosts。
>
> 在 Linux 上则应查看/etc/hosts。

实际上，这个场景非常普遍。恶意软件在几年前就使用这个方法，把用户重定向到存放恶意代码的网站。试想，如果黑客修改了你的 hosts 文件，每次你登录网上银行，实际上访问的却是一个伪造的网站，专门偷你账户里的钱，这该有多恐怖！

3．学到的知识

继续分析流量，你会了解各种各样的协议如何工作以及如何阻断它们。在这个场景中，主机没有发送 DNS 请求是因为客户端被错误配置了，而不是因为其他外部限制或外部的错误配置。

在数据包层面上考查这个问题，我们可以迅速发现未知的 IP 地址，也能迅速发现 DNS 这个通信过程的关键部分消失了。通过这些信息，我们可以指出客户端才是问题的来源。