

邮件伪造之SPF绕过的5种思路

原创 Bypass Bypass 今天

SMTP(SimpleMail Transfer Protocol) 即简单邮件传输协议，正如名字所暗示的那样，它其实是一个非常简单的传输协议，无需身份认证，而且发件人的邮箱地址是可以由发信方任意声明的，利用这个特性可以伪造任意发件人。

SPF出现的目的，就是为了防止随意伪造发件人。SPF，全称为 Sender Policy Framework，是一种以IP地址认证电子邮件发件人身份的技术。邮件接收方首先会去检查域名的SPF记录，来确定发件人的IP地址是否被包含在SPF记录里面，如果在，就认为是一封正确的邮件，否则会认为是一封伪造的邮件并进行退回。

众所周知，如果没有配置SPF，攻击者可以任意伪造邮件，即使配置了SPF，在特定的情况下，依然可以完美伪造邮件发件人。当我们开始查看一个目标邮箱的SPF记录时，一场关于邮件安全策略的对抗也就开始了。

```
1 nslookup -type=txt qq.com
```

本文结合SPF配置的过程，通过一些邮件测试验证，分享了5种SPF绕过的思路。从攻击者的视角出发，来看看它会如何绕过SPF检测，完美伪造邮件人地址，并成功投递到目标邮箱，欢迎指正和补充~

1、SPF解析不当导致绕过

假设我的SPF记录设置为：

```
1 v=spf1 ip4:220.xxx.10.0/24 ~all
```

这条SPF记录的意思是说只允许 220.xxx.10.1~220.xxx.10.255 范围内的IP，软拒绝，发件 IP 非法，但是不采取强硬措施。

这就存在两个严重的安全隐患：

一个是IP段过大，在C段里面，只要获取任意一台主机的权限，那么就可以使用合法的IP进行邮件伪造。

一个是软拒绝，也就是会接受来信，但可能被标记为垃圾邮件。如果SPF记录设置拒绝，就会有大量的邮件被丢弃或者隔离，影响办公效率，有一些邮件系统管理员为了减少业务影响，而采用软拒绝的策略。

当SPF记录设置成~all时，通过测试可以发现，outlook邮箱可以接收邮件，QQ邮箱不接收，163邮箱被标记为垃圾邮件。

还有一种极为严重的错误，就是SPF解析记录配置错误，早在之前鹅厂就出现过SPF解析错误，比如：

```
1 v=spf1 ip4:113.110.223.0/24 183.110.226.0/24 183.110.255.0/24 59.110.132.0/24
```

这里介绍一个工具，输入域名和SPF记录，可快速检查SPF记录是否正确

测试地址：


```
1 https://www.kitterman.com/spf/validate.html
```

Input accepted, querying now...

evaluating v=spf1 ip4:113.110.223.0/24 183.110.226.0/24 183.110.255.0/24 59.110.132.0/24 -all ...

Results - PermError SPF Permanent Error: Missing IP4: 183.110.226.0/24

[Return to SPF checking tool \(clears form\)](#)

Use the back button on your browser to return to the SPF checking tool without clearing the  微信号: Bypass--

SPF记录报错，在这条SPF记录中，存在多个IP段，但只有开头的一段ip用了ipv4，这就导致了语法错误。因为这个错误，将导致整个SPF记录完全失效，因为SPF无效，邮件接收方的SPF检测功能也就失效了。

综上，当我们在查看一个域名的SPF记录时，它其实不只是一条解析记录，更是一种邮件安全的策略，SPF记录配置不严或SPF解析错误，就容易导致大量本该被拦截的邮件直接被放进来，而绕过的策略就隐藏在这条SPF记录里面。

2、SPF配置不当导致绕过

邮件服务器管理员做SPF配置时，其实是需要两个步骤的，首先在域名中增加SPF记录，向支持SPF功能的邮件服务器提供验证信息，使别人能验证自己；另外，需要配置邮件服务器支持 SPF，这样才可以验证别人。

那么，在SPF配置过程中，也常常因为配置不当导致绕过，比如：

第一种情况：

域名增加了SPF记录，但是邮件服务器不支持SPF检查或邮件网关未开启SPF检测，无法验证邮件来源。这种情况下，我们声明了自己是谁，但却无法验证对方是谁，SPF检测无效，可伪造任意用户发送到你的域名邮箱里。

第二种情况：

SPF解析在公网DNS，邮件服务器配置内部DNS，内部DNS无法进行SPF解析，从而导致绕过，可从公网伪造任意用户发送邮件。

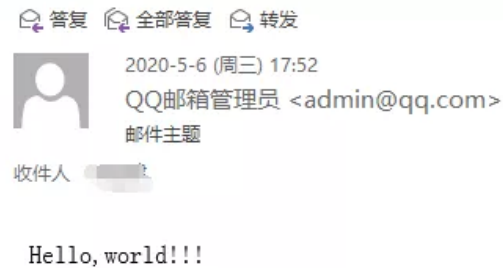
第三种情况：

攻击者在公司内网，内网SMTP服务器开启匿名邮件发送或者在信任中继服务器IP段，就可以使用任意用户发送邮件。

比如，当 mynetworks = 192.168.0.0/16 ，在内网，任意一台终端就可以直连公司的SMTP服务器，伪造了一封来自 admin@qq.com 的邮件发给自己。

```
1 python SimpleEmailSpoofers.py -t [目标邮箱] -n QQ邮箱管理员 -f admin@qq.com -j '
```

测试效果如下：



微信号: Bypass--

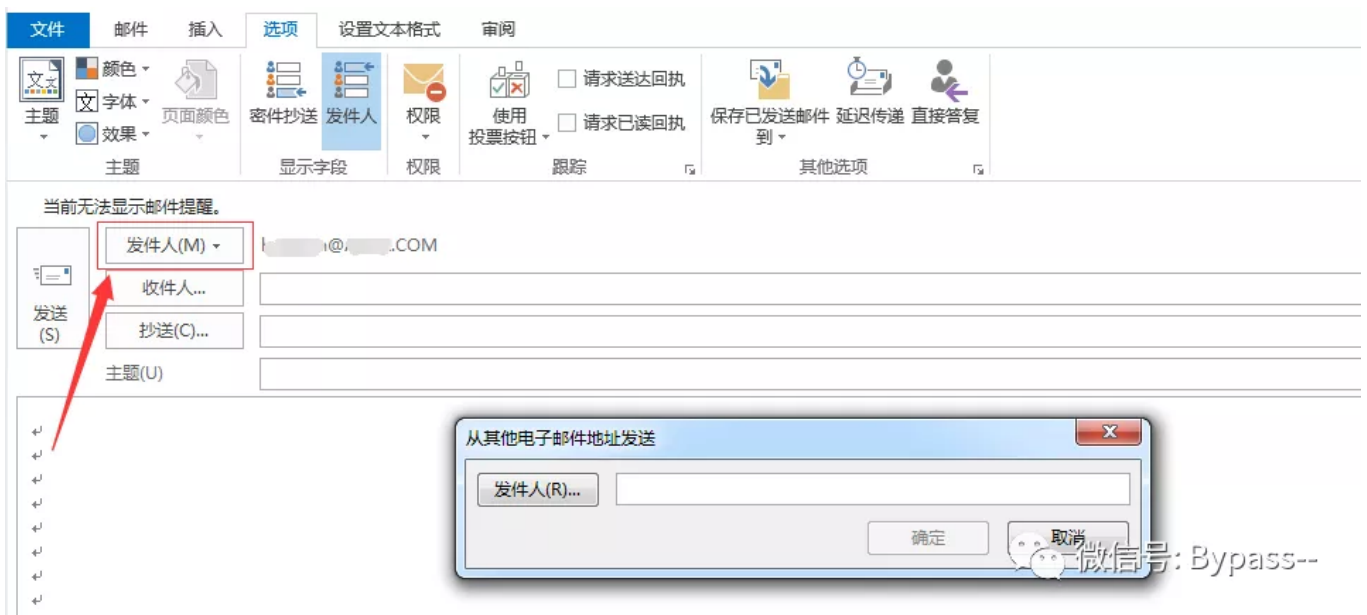
3、高权限用户绕过

对于Exchange邮箱系统，拥有 Domain admin 权限的域用户，可通过outlook直接指定发件人，伪造任意发件人发送邮件。伪造邮件的方式十分简单，且邮件头无法显示真实IP。

测试过程：我给自己的账号也添加了 Domain admin 权限。

使用 Outlook2013 客户端指定发件人发送邮件，接收邮件直接显示伪造人的名字，伪造成功。

使用 Outlook2016 客户端测试，邮件接收方的发件人位置显示“XXX代表XXX”。



存在一定的邮件伪造风险，但在实际中意义并不大，如果拥有了 Domain admin 权限，哪里还需要邮件伪造呢？

4、邮件客户端内容解析差异

很多时候，大部分的企业邮箱SPF配置都是正确的，理论上，它会对每一封邮件进行检测，那么它是怎么验证发件人的IP地址的呢？

我们使用一个SPF在线检测的工具，来做一下小小的尝试，利用我本地搭建的匿名SMTP服务器伪造admin@qq.com邮箱。

测试地址：

```
1 https://www.kitterman.com/spf/validate.html
```

- 1、在IP address: 里输入将要发信的IP地址，即本地ip地址。
- 2、SPF Record v=spf1...: 输入nslookup查出来的SPF记录
- 3、Mail From address: 输入将要发信的发件人

Test an SPF record

This test is for evaluating the performance of your record based on different IP addresses that mail might come from (this is the IP address of the mail server). It can also be used for syntax checking of records with more complex macros (although this has not been thoroughly tested yet). The %d macro is extracted from the supplied mail from address. If an SPF record is supplied, it is used for the initial evaluation instead of any record published in DNS for the domain.

IP address:

SPF Record v=spf1...:

Mail From address:

HELO/EHLO Address:

微信号: Bypass--

点击Test SPF Recod进行验证：

Input accepted, querying now...

Mail sent from this IP address: 120.36.251.149
Mail from (Sender): admin@qq.com
Mail checked using this SPF policy: v=spf1 include:spf.mail.qq.com -all
Results - FAIL Message may be rejected

[Return to SPF checking tool \(clears form\)](#)

Use the back button on your browser to return to the SPF checking tool without clearing the form.

微信号: Bypass--

结果毫无疑问，SPF验证失败，伪造邮箱不成功，伪造的邮件将会被退回。

通过查看邮件头信息，有两个比较重要的字段，Sender和From。

Sender字段，代表的是邮件的实际发送者，邮件接收方会对它的邮件域名进行SPF检测，确认是否包含了发信人的IP地址。From字段，代表的是邮件发送人，即邮件里所显示的发件人，容易被伪

造。

在SPF配置有效的情况下，Sender必须通过SPF检验，所以我们可以设置为正常的邮件服务器地址，然后对From字段进行伪造。

使用swaks做一个邮件测试：

```
1 sudo ./swaks --to 67*****28@qq.com --from admin@evil.com --h-From: '=?GB2312:
2
3 其中参数：
4 --from <实际发件人，对应Sender字段>
5 --h-From <邮件显示的发件人，对应From字段>
```

QQ邮箱网页版查看邮件，Sender和From字段不一样时，发件人的位置显示由admin@evil.com代发。



使用Foxmail客户端查看同一封邮件，Sender和From字段不一样时，不显示代发，伪造成功。



我们分别使用网页版邮箱和客户端邮箱打开同一封邮件，通过对比可以发现，不同的邮件客户端对发件人位置的内容解析是不一样的。

平时工作中，不少使用腾讯企业邮箱的童鞋，都喜欢使用Foxmail客户端查收邮件，这就给了我们成功伪造邮件的可乘之机。

通过测试可以发现：qq邮箱、163邮箱网页版均会显示代发，Outlook邮箱不显示代发，具体邮件客户端软件可具体再行测试。

5、From字段名截断绕过

当我们伪造邮件发送成功的时候，由于Sender和From字段不一样，部分邮件客户端接收邮件后，会提示邮件代发。



那么有没有办法只显示伪造的发件人，不显示邮件代发呢？

在网络上看到一种思路，来源于网贴《关于邮件伪造的一些新思路》，挺有意思的。

在用SMTP发送电子邮件时，发件人别名，格式为：From：发件人别名<邮件地址>。通过对发件人别名字段填充大量的特殊字符，使邮箱客户端截取真实的邮件地址失败，从而只展示我们伪造的发件人别名和伪造邮箱。

邮件伪造测试过程：

1、在QQ邮箱中导出mail.eml文件，删除前面不必要的字段信息。

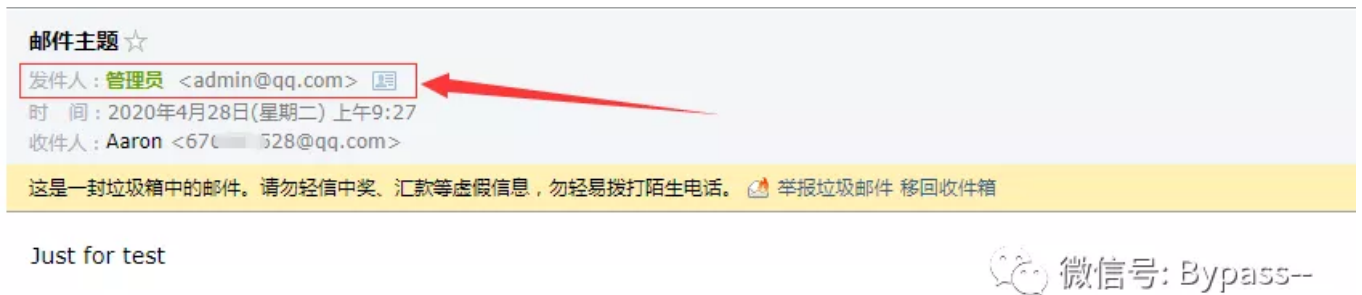
2、填充发件人别名，伪造邮件头Fron字段：

```
1 From:=?gb2312?B?udzA7dSxIDxhZG1pbkBxcS5jb20+0aGhoaGhoaGhoaGhoaGhoaGhoaGhoQ==?:
2 =?gb2312?B?oaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGh?=
3 =?gb2312?B?oaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGh?=
4 =?gb2312?B?oaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGh?=
5 =?gb2312?B?oaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGhoaGh?=
6 =?gb2312?B?oaGhoaGhoaGhoaGhoaGhoaGhoaGhoSAgICAgICAgICAgICAgICAgIKGkoaQ==?:
7 =?gb2312?B?oaQgICAgICAgICAgICAgICAgIKGhICAgICAgIKGkoaShpA==? <admin@test.com>
```

3、使用 —data参数发送邮件。


```
1 sudo ./swaks --data mail.eml --to 67*****28@qq.com --from admin@test.com
```

4、成功发送给目标邮箱，QQ邮箱接收邮件后的呈现效果：



备注：从测试情况看，我伪造的邮件进了QQ垃圾箱，但这种思路还是挺不错的，重新Fuzz，或许可以构造特殊的数据包触发这个问题。