

原文：

<https://laconicwolf.com/2018/09/21/mfa-bypass-and-privilege-escalation/>

本文主要讲解两个MFA(多因素验证)应用在web应用程序中部署实现时存在的问题。MFA的确是非常好的，我非常支持它，但在很多应用系统中，第二个因素的验证与第一个因素

1. 应用一 Okta：第二个因素绕过，没有帐户锁定

该系统允许自行注册，并使用Okta强制执行身份验证。你可能不知道Okta是什么，它就是一个身份和访问管理解决方案，可以集成到你的应用程序中，以提供集中管理和


我们先来走一遍该系统的正常身份验证流程，然后再来讨论这些漏洞。


正常身份验证流程

Login


We use two-factor authentication to protect your account. This includes strict password requirements and a telephone call you will receive when you attempt to log in.


Username

 DUTN




Password





[Forgot Password?](#)

 LOG IN



使用用户名和密码登录，如图：

Target	Proxy	Spider	Scanner	Intruder	Repeater	Sequencer	Decoder	Comparer	Ex
Intercept	HTTP history	WebSockets history	Options						
Request to https:// [REDACTED]									
Forward	Drop	Intercept is on	Action						
Raw	Params	Headers	Hex	Decoded Authorization Header	JSON Beautifier				

POST /api/auth/v1.0.0/auth/authenticateOktaUser HTTP/1.1

Host: [REDACTED]

Connection: close

Content-Length: 54

Accept: application/json, text/plain, */*

Origin: https:// [REDACTED]

Authorization: Basic [REDACTED]

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/68.0.3440.106 Safari/537.36

Content-Type: application/json

Referer: https:// [REDACTED]/login

Accept-Encoding: gzip, deflate

Accept-Language: en-US,en;q=0.9

```
{
  "username": "DUTN",
  "password": "[REDACTED]"
}
```

先知社区

服务器检查用户名/密码是否正确，返回第二个因素要提交的必要数据，这些数据具体来说就是状态令牌，oktaUserId和factorId：

```
{
  "stateToken": "00DsAt [REDACTED] bNL3",
  "expiresAt": "2018-08-16T19:27:39.000+0000",
  "status": "MFA_REQUIRED",
  "oktaRecoveryToken": null,
  "oktaUserId": [REDACTED],
  "passwordChanged": "2018-08-14T15:29:52.000+0000",
  "factorID": "sms [REDACTED]",
  "factorType": "sms",
  "errorSummary": null,
  "qrCode": null,
  "qrCodeType": null,
  "securityQuestion": null
}
```

先知社区

在上述响应之后，将自动提交一个请求以启动第二个因素的SMS通知，然后你会看到如下的提示：

Login

We use two-factor authentication to protect your account. This includes strict password requirements and a telephone call you will receive when you attempt to log in.



Enter Code 284907

✓ VERIFY

Didn't receive the code, or it expired, [click here to resend](#).



提交第二个因素并确认后，会自动发起第二个身份验证请求，如图：

TargetProxySpiderScannerIntruderRepeaterSequencerDecoderComparer

InterceptHTTP historyWebSockets historyOptions

Request to https://[REDACTED]

ForwardDropIntercept is onAction

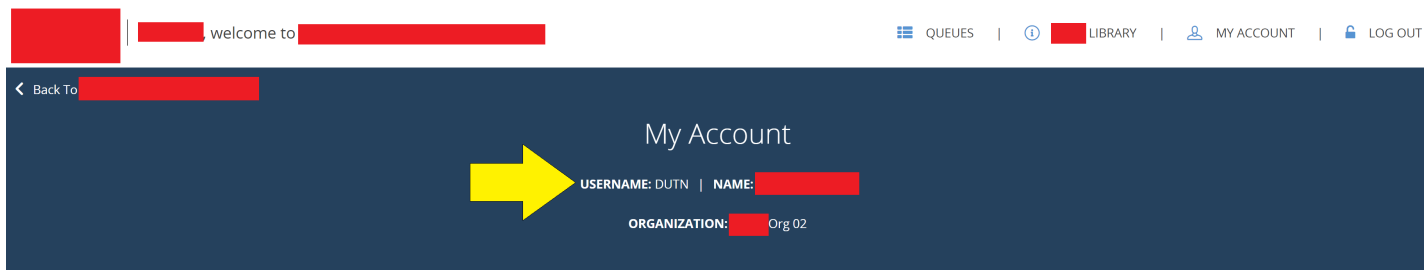
RawParamsHeadersHexDecoded Authorization HeaderJSON Beautifier

POST /api/auth/v1.0.0/auth/oktaToken HTTP/1.1
Host: [REDACTED]
Connection: close
Content-Length: 54
Accept: application/json, text/plain, */*
Origin: https://[REDACTED]
Authorization: Basic [REDACTED]
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/68.0.3440.106 Safari/537.36
Content-Type: application/json
Referer: https://[REDACTED]/login
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9

{
 "username": "DUTN",
 "password": "[REDACTED]"
}



服务器验证你的凭证，如果凭证有效，会返回JWT（json web token）响应，这样你就可以访问应用程序了。



漏洞

使用MFA的应用对无效的用户名/密码的不断尝试并没有严格的账户锁定策略。这个应用只是通过JavaScript向你显示锁定警告，但在服务器后端却没有进行逻辑验证，刷新

没有账号锁定就太好了，不过，即使我们可以暴力破解出密码，我们仍然要考虑第二个因素。幸运的是，该应用对MFA和身份验证请求是独立验证的。

于是我就执行了下列操作：

没有帐户锁定，我能够暴力破解出另一个用户的密码。我已经在该系统上有一个有效的帐户（可以自行注册），所以我用初始用户名/密码进行登录（服务器返回了stateToken），然后拦截第二个用户名/密码身份验证请求，进行暴力破解得到正确的用户和密码，替换掉我的用户名和密码，如图：

Target	Proxy	Spider	Scanner	Intruder	Repeater	Sequencer	Decoder	Comparer
--------	-------	--------	---------	----------	----------	-----------	---------	----------

Intercept HTTP history WebSockets history Options

Request to https://[redacted]

Forward Drop Intercept is on Action

Raw Params Headers Hex Decoded Authorization Header JSON Beautifier

POST /api/auth/v1.0.0/auth/oktaToken HTTP/1.1

Host: [redacted]

Connection: close

Content-Length: 54

Accept: application/json, text/plain, */*

Origin: https://[redacted]

Authorization: Basic [redacted]

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/68.0.3440.106 Safari/537.36

Content-Type: application/json

Referer: https://[redacted]/login

Accept-Encoding: gzip, deflate

Accept-Language: en-US,en;q=0.9

```
{
  "username": "mAAK",
  "password": "[redacted]"
}
```

然后，服务器返回了一个JWT响应，这样我就能够以第二个用户的身份访问应用了，如图：

Contact Information

EDIT

EMAIL

CONTACT PHONE NUMBER

CONTACT PHONE EXTENSION



1. 应用二 Symantec VIP：提权到任意用户

这个漏洞更严重。该应用没有自行注册功能，但我和团队每个人都有两个帐户（一个普通用户，一个管理员）进行测试。该系统使用Symantec VIP验证第二个因素。同样，使用Symantec VIP本身没有任何问题，但在这个案例中，实现方式是有问题的。我们先还是来走一遍正常的验证流程，然后再来讨论其中的问题。

正常身份验证流程

使用用户名和密码登录，如图：

Welcome to

NOTE: Fields marked with an asterisk (*) are required.

* Login:

pentest_jake

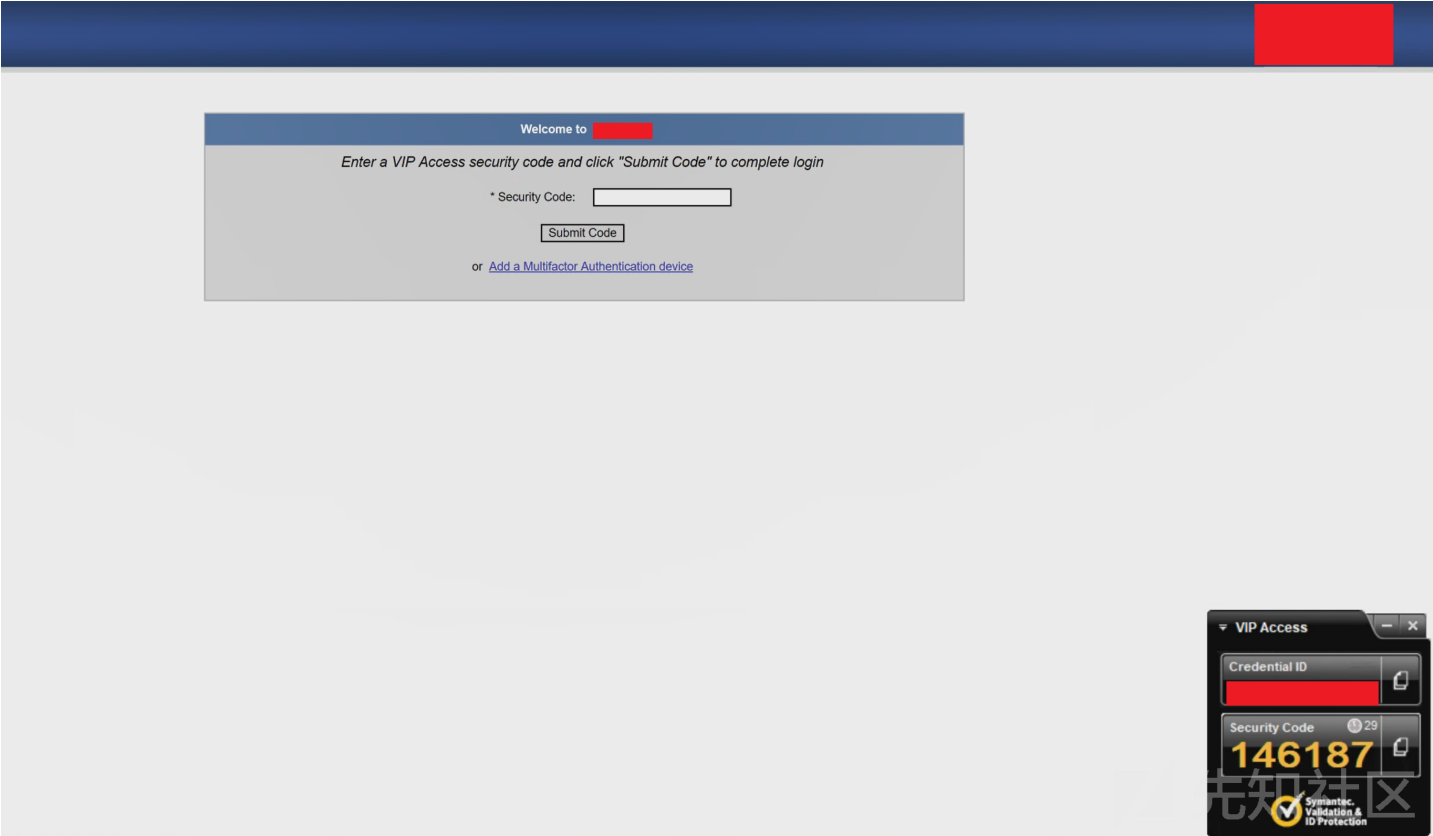
* Password:

Next

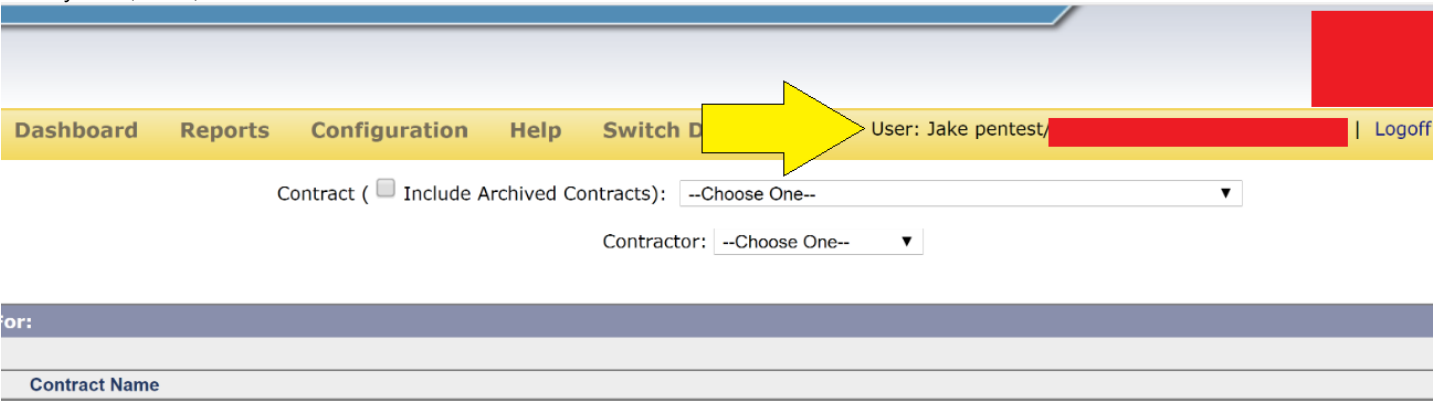
服务器检查用户名和密码，返回提交和验证第二个因素所必需的数据，也就是hidUserId和hidMfaId，它们只不过是一个防篡改的查询字符串值，实际上是一个带有签名Mjc3OA== - 5CBeGaUhkiA =

' - '的左侧解码为2778，' - '的右侧是签名，但即使我想更改该值并重新编码，签名也不正确，我会得到一个错误。hidUserId只是一个数字，而hidMfaId只是用户名，但两者都使用防篡改查询字符串

使用用户名密码登录后，hidUserId和hidMdaId值存在于响应包中，然后进入到下一步，需要我进行第二个因素验证，如图：



security code(安全码)与hidUserId和hidMfaId一起提交，如果安全码正确，则授予访问权限，如图：



漏洞
与第一个系统类似，此应用程序将MFA作为两个独立的步骤来处理。第一个请求验证用户名/密码并返回了防篡改的hidUserId和hidMfaId，第二个请求验证了安全码。我注
凑巧的是，这个应用程序还有另一个漏洞，让我能够获得该值。
下面就是我所操作的：
如果我使用一个有效用户名和无效密码进行登录，那么应用程序就不会返回hidUserId，但是状态会改变，当我再次尝试登录时（使用相同的有效用户名和无效密码），hidU

POST request to [redacted]logon.aspx

Type	Name	Value
URL	ReturnUrl	[redacted]
Cookie	ASP.NET_SessionId	
Body	__EVENTTARGET	
Body	__EVENTARGUMENT	
Body	__VIEWSTATE	
Body	__VIEWSTATEGENERATOR	
Body	__EVENTVALIDATION	
Body	txtUserName	pentest_jake
Body	txtUserPass	asdf
Body	btnSubmitCredentials	Next
Body	hidPasswordDate	
Body	hidUserId	Mjc3Mg==b8yUddqrN+Y=
Body	hidMfaldentity	

现在测试是否可以在普通用户的有效身份验证请求中替换成admin用户的hidUserId。所以我再次使用我的普通用户帐户pentest_jake，并使用用户名和密码登录，如图：

Welcome to [redacted]

NOTE: Fields marked with an asterisk (*) are required.

* Login:

* Password:

Next

然后我提交了我的安全码并拦截请求，用管理员帐户UserId替换了我的hidUserId值，如图：

Target Proxy Spider Scanner Intruder Repeater Sequencer Decoder Comparer Extender Project options User options Alerts CO2

Intercept HTTP history WebSockets history Options

Request to https://[redacted]

Forward Drop Intercept is on Action

Comment this item

Raw Params Headers Hex ViewState

POST request to [redacted]logon.aspx

Type	Name	Value	
Cookie	ASP.NET_SessionId		Add
Body	__EVENTTARGET	[redacted]	Remove
Body	__EVENTARGUMENT		Up
Body	__VIEWSTATE		Down
Body	__VIEWSTATEGENERATOR		
Body	__EVENTVALIDATION		
Body	txtUserName	pentest_herman_admin	
Body	txtUserPass	asdf	
Body	btnSubmitCredentials	Next	
Body	hidPasswordDate		
Body	hidUserId	Mjc3OA==5CBeGaUhkIA=	
Body	hidMfaldentity		

这样，我就以admin用户登成功录到应用了，如图：

Secure | https://[redacted]home.aspx

Home Dashboard Reports Configuration Help Admin News and Updates Switch Dept

User: herman admin pentest/[redacted] Logoff TEST MODE IS ON

--Important News--

NEW! 7/16/2018: HI

HI

最后，我重申一下，我绝对支持MFA，但实现起来可能会出现问題，而且这些问題非常常见。

点击收藏 | 0 关注 | 1
[上一篇：Vulnhub - Bob 靶机渗透攻略](#)
[下一篇：提高AFL qemu模式性能](#)

1. 0 条回复
- 动动手指，沙发就是你的了！

[登录](#) 后跟帖

先知社区

[现在登录](#)

热门节点

[技术文章](#)

[社区小黑板](#)

目录

[RSS](#)
[关于社区](#)
[友情链接](#)
[社区小黑板](#)