SSH BRUTEFORCE PASSWORDS

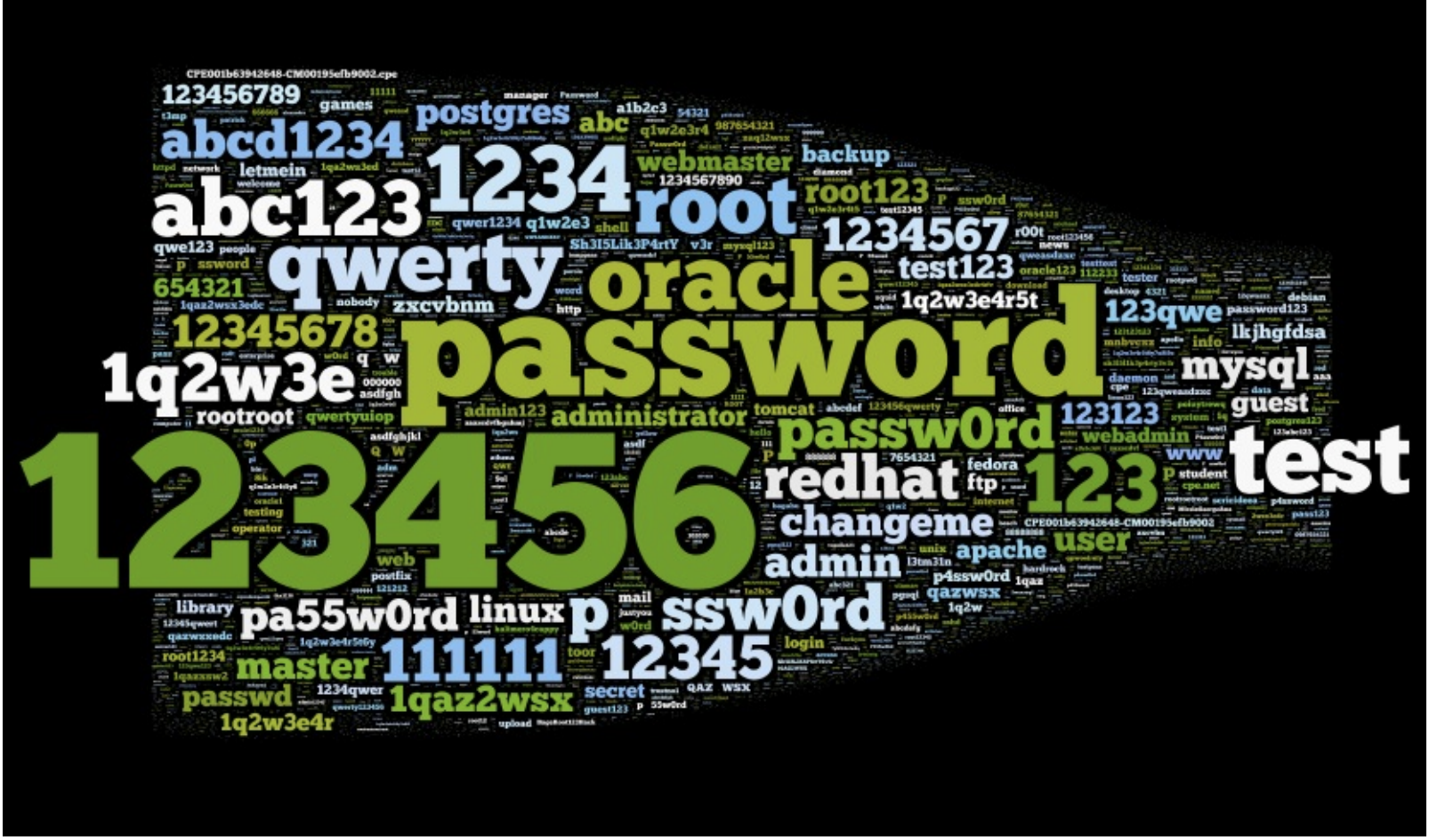
For the last almost two months (46 days) I have been logging the attempts of those trying to bruteforce my little ssh daemon. At first it was more about could I do it, but as the logs grew, it was interesting to see what else could be seen. So what did I find out?

```
Total number of passwords : 36968
Passwords 10 or more characters : 3479
Passwords 6 or less characters : 11797
Passwords 3 or less characters : 1151
Passwords with special characters : 1574
Passwords with only numerics : 2338
Passwords with only alpha characters : 11496
Non-honeypot user attempts : 13255
```

Firstly, that is an average number of 803 attempts a day. I did not expect that on my little server. Since I only have 64 honeypot users which log passwords that means those 64 accounts where attempted 23713 times. Of those only 1574 passwords used special characters, and only 3479 were passwords of 10 characters or more. So we can quite definitely see that the bruteforce attempts are aimed at common mistakes and low hanging fruit.

```
Top 10 attempted passwords :
    125 p@ssw0rd
    136 1q2w3e
    148 root
    157 abc123
    160 1234
    168 qwerty
    171 oracle
    187 test
    294 password
    436 123456
```

The list of the top 10 passwords supports the above conclusions. Just look at the top 5 attempted passwords, anyone using those.. well I question whether they should be looking ater servers. Lets take a look at the data another way, I created a 'tag cloud' from the full password list..



This shows us that most of the commonly used passwords (the largest 'tags') are very simple, thus very bad, passwords. Lets dig down a little, 'password' was attempted 294 times, but what about 'password' and all it's derivatives?

```
Versions of 'password' :
      1 p4sSw0rd
      1 p4Ssw0rd
      1 PaSsWoRd
      1 PASSword
      1 pa$$w0rd
      1 p@sSw0rd
      1 p@Ssw0rd
      2 Pa55w0rd
      2 Pa55word
      2 pa$$word
      5 PASSW0RD
      9 p455word
      9 p455w0rd
      9 P455word
      9 P455w0rd
      9 p4ssw0rd
      9 P4ssw0rd
      9 p@55w0rd
      9 P@55w0rd
      9 passw0rd
      9 p@ssw0rd
      9 P@ssw0rd
      9 p@$$word
      9 p@$$w0rd
      9 P@$$word
```

```
      9 P@$$w0rd
     10 P455w0rd
     10 P4ssword
     10 p@55word
     10 pa55word
     10 P@$$w0rd
     11 P@55word
     11 Passw0rd
     12 P4ssw0rd
     12 P@55w0rd
     12 P@ssword
     14 p@$$w0rd
     15 Passw0rd
     16 Password
     17 p455w0rd
     18 p4ssword
     19 p@55w0rd
     27 P@ssw0rd
     29 p@ssword
     32 p4ssw0rd
     88 pa55w0rd
    115 passw0rd
    125 p@ssw0rd
    294 password
Total: 1070
```

Here we see that the bruteforcers tried many variations of password, giving a total account for the term 'password' of 1070, which will easily make it the most attempted password (pardon the pun). This also shows us rather blatantly that trying to make a bad password better does not work. We see similar trends in other attempts..

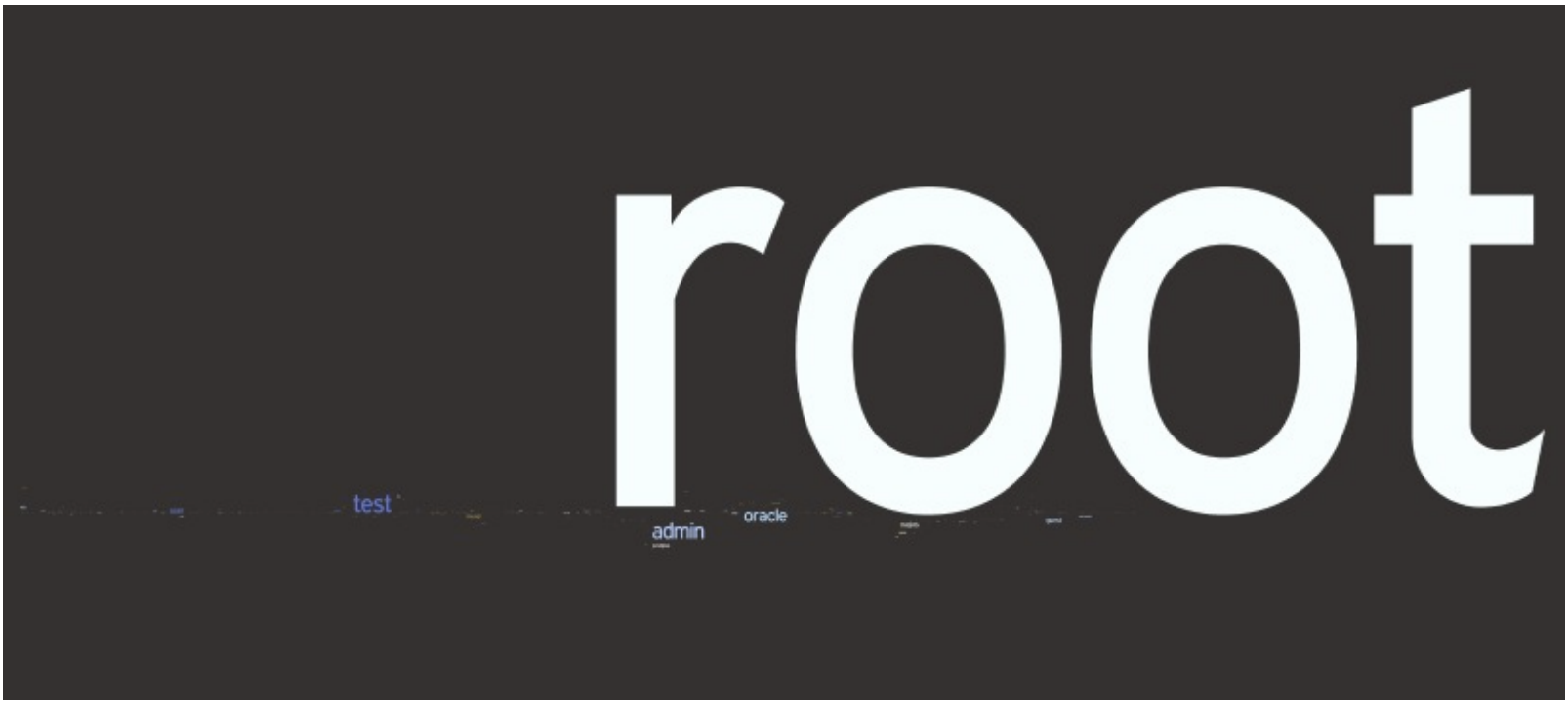
```
Versions of 'test' :
      1 Test
      9 t3st
    187 test
Total: 197

Versions of 'oracle' :
      3 0r4cl3
      3 or4cl3
      6 0racle
      7 0racl3
      8 oracl3
    171 oracle
Total: 198
```

But nothing quite as severe, although we see once again that substitution attempts in passwords do not work. But what about the other side of the password combination, the username?

```
Top 10 attempted usernames :
    122 web
    157 postgres
    201 mysql
    216 nagios
    230 guest
    242 user
    574 oracle
    703 admin
```

```
  784 test
18882 root
```

This was quote something. Attempts on the 'root' user account for just over 50% of all attempts. Adding together the occurences for the other 9 in the top 10 list still does not equal the attempts made on root. I created another 'tag cloud' or the username data which graphically shows the situation rather well..

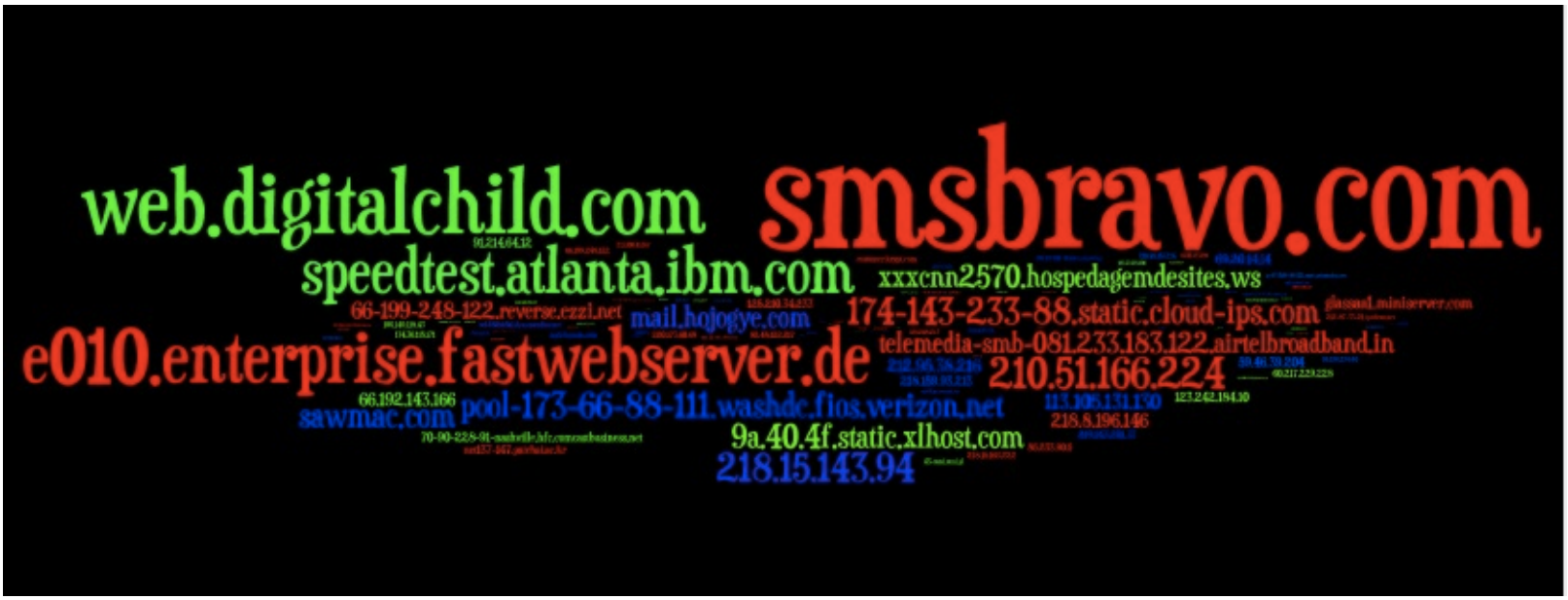


Yep, fairly obvious that 'root' is what everyone is after. But while that graphic makes my point about 'root' it does not help us analyze the other attempts very well, so I created another 'tag cloud' and excluded the 'root' attempts..

Well, obviously the bruteforcers think oracle, postgres, mysql administrators do not look after their accounts very well. It is easy to see that the majority of non-'root' username attempts are still commonly used accounts. If you see your account here, give it a good password please. But what about the source of these attempts?

```
 964 9a.40.4f.static.xlhost.com
1042 pool-173-66-88-111.washdc.fios.verizon.net
1056 sawmac.com
1060 174-143-233-88.static.cloud-ips.com
1145 218.15.143.94
1240 210.51.166.224
1876 speedtest.atlanta.ibm.com
2199 e010.enterprise.fastwebserver.de
2596 web.digitalchild.com
4734 smsbravo.com
```

In the top ten list, only 2 do not have a resolvable name of some kind, that means of the other 8 have had someone do some work on them. And the names are fairly interesting as well. Look at the domains. And once again, here is a 'tag cloud' for the source data..



Final Words

So in the end what can we see? If you do not allow 'root' access that is over 50% of all attempts straightaway. Choosing a decent password (10 characters or more with special characters) would negate over 75% of all attempts. There are of course other things you can do, limit what a user can run, limit from where logins can come, use certificates and many more - but what we see is that just some simple -very simple- basic administration tasks will negate almost every bruteforce attempt.