Password Algorithms: K9 Web Protection Admin

Posted on **June 30, 2012** by **dietrich**

Introduction

K9 Web Protection by Blue Coat software is a cheap but effective solution for parents monitoring internet activity of their children.
It's also a cheap solution for internet cafes and companies to monitor customers and staff.

According to the vendors website, it's installed on some 3 million computers and presumably most of those are using the freeware version for home use which is what I've downloaded.

Although there's a commercial version, I'm not aware of any differences between the 2 with respect to the password algorithm.

Storage Method

Installed on XP, most files are kept in:

```
C:\Program Files\Blue Coat K9 Web Protection
```

The **license** file is always 944 bytes in size and also where the Admin password is stored.

Here's the actual password "hash" and offset of where it appears.

```
000001d0: c4 b8 b5 b5 b7 b7 bd b1 - be 29 6b d6 eb 2c a9 00
```

The offset was discovered through analysing one of the binaries.

So how is this “hash” generated? 🙂

You can recover this password with simple subtraction. (at least in my own case) The example you see above is ***“theeggman”*** and will be revealed if you subtract **0×50** from each byte until you reach an invalid character.

The subtraction value may be different for you since I haven’t checked beyond my own XP install. However, it’s possible to get the value from the license file from offset **0x2c**

```
00000020: 75 a5 ce 51 00 00 00 00 - c3 e1 5a 78 50 00 00 00
```

Generation

Below is rough code for how a password is generated.

```
#define MAX_PASS_LEN 15

void encode(unsigned char k9_hash[], char password[]) {
    size_t pass_len = strlen(password);
    size_t i;

    for (i = 0;i < pass_len && i < MAX_PASS_LEN;i++) {
      k9_hash[i] = password[i] + license_data[44];
    }

    for (;i < MAX_PASS_LEN;i++) {
      k9_hash[i] = rand() % 256;
    }
    k9_hash[MAX_PASS_LEN] = 0;
}
```

Recovery

This uses hardcoded values from my own license file and successfully decodes to plaintext.

```
#include <stdio.h>
#include <ctype.h>

#define MAX_PASS_LEN 15

char* decode(char password[], unsigned char k9_hash[]) {
  char sym[] = "!@#$%^*(){}";
  size_t pass_len = 0;

  for (;pass_len < MAX_PASS_LEN;pass_len++) {
    int c = k9_hash[pass_len] - 0x50;
    if (!isalnum(c)) {
      if (!strchr(sym, c)) {
```

```
        break;
      }
    }
    password[pass_len] = c;
  }
  password[pass_len] = 0;
  return password;
}

unsigned char k9_admin_pass[] = { 0xc4, 0xb8, 0xb5, 0xb5,
                                  0xb7, 0xb7, 0xbd, 0xb1,
                                  0xbe, 0x29, 0x6b, 0xd6,
                                  0xeb, 0x2c, 0xa9, 0x00 };

void main(void) {
  char password[MAX_PASS_LEN+1];

  memset(password, 0, sizeof(password));
  printf("\nPassword = %s\n", decode(password, k9_admin_pass));
}
```

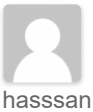
Conclusion

In order to take advantage of the weak hashing algorithm, you need at least read access to the license file and after checking permissions, it's only available to the owner, SYSTEM and other local administrators.

It doesn't provide much protection for the password but controlling admin access at OS level negates this.

This entry was posted in **Algorithms**, **Passwords**, **Reverse Engineering**, **Security** and tagged **algorithms**, **assembler**, **bluecoat**, **C#**, **hashing**, **K9**, **math**, **monitoring software**, **parental monitoring software**, **password algorithm**, **passwords**, **Reverse Engineering**, **web protection** by **dietrich**. Bookmark the **permalink [https://web.archive.org/web/20140207115505/http://insecurety.net/?p=148]** .

4 THOUGHTS ON "PASSWORD ALGORITHMS: K9 WEB PROTECTION ADMIN"



hasssan

on **August 17, 2012 at 7:30 pm** said:

Error 1 error C2065: 'license_data' : undeclared identifier c:\users\hp\documents\visual studio 2010\projects\gen\gen\gen_s.cpp 12 1 gen

dietrich

on **August 17, 2012 at 9:55 pm** said:

Hello,

license_data is buffer filled by fread()

The algorithm is really simple and can be solved using calc.exe and a hex dumper.

Sorry I can't provide full source code at this time but enough information is available to develop your own.

Michel Ruisch
on **September 10, 2013 at 9:43 pm** said:

i don't understand… because i opened in Hex dump tool. and used calc.exe. what do you mean 0×50? i trying here but didn't retrive my password. I have installed K9 for tests. using "wxHexEditor"

Michel Ruisch
on **September 10, 2013 at 10:27 pm** said:

what software you use to run the algorithms