IPv6 Attack

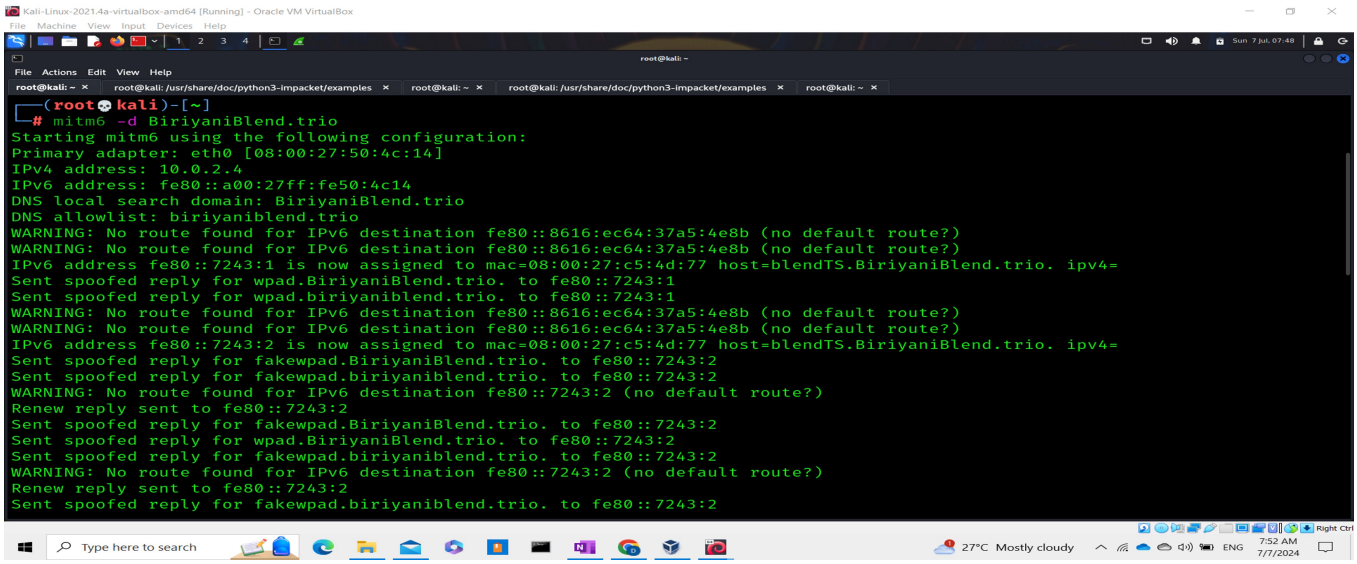
Saturday, June 29, 2024 3:05 AM

Attack Description:

The python based tool 'mitm6' by responding to the DHCPv6 queries on the specified domain and by periodically sending router advertisement spoofs it self as the IPv6 DNS on the network. Also stars spoofing for any wpad query and starts setting the attacker machine as the proxy server to connect and authenticate from in the query request for /wpad.dat. Used together with ntlmrelayx.py which serves the fake wpad host to requesting machine with -wh parameter and with it specify the host that the WPAD file resides on. Since mitm6 gives us control over the DNS, any non-existing hostname in the victim network will do.

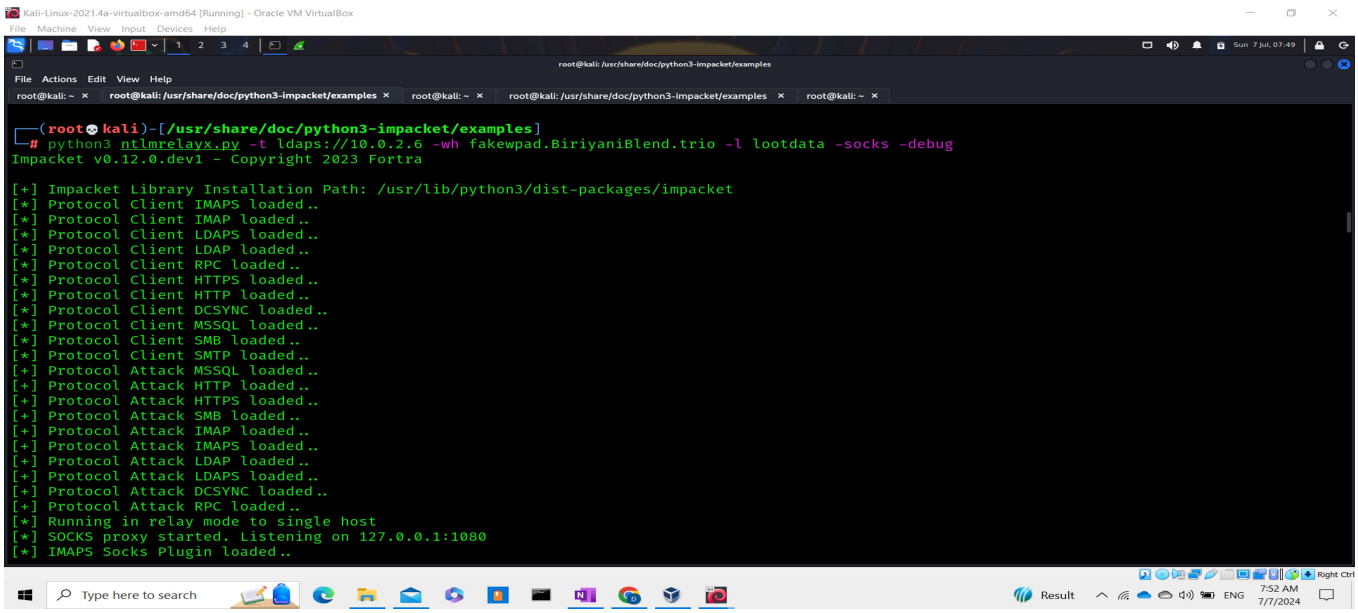
Attack Flow:

Step 1: Execute mitm6



-d flag is used to specify the domain to tell the tool to spoof reply to the machines belonging to this domain

Step 2: In another tab execute ntlmrelayx.py

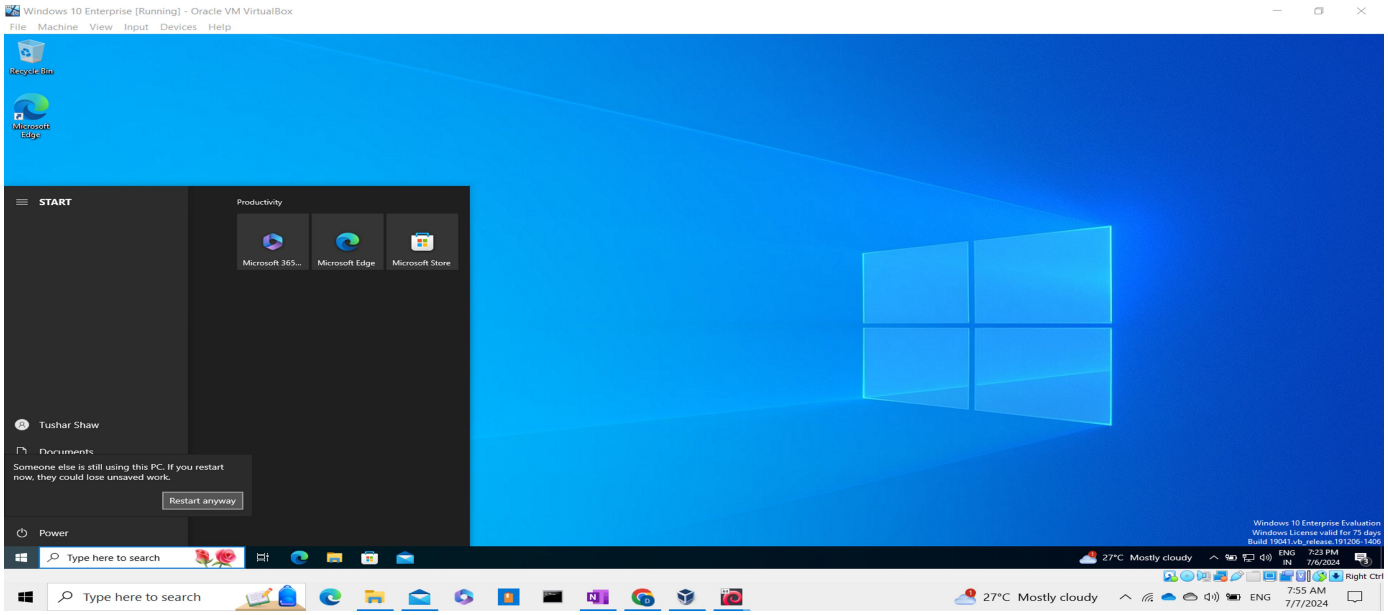


-t specifies the target service and the ip
-wh specifies the fake name for wpad in the domain
-l file to store all the exfiltrated data once login through the service

Now from here in an actual pentest environment we have to wait till some event off rebooting a pc and a user login on to the workstation, which will trigger DHCP v6 broadcast in the local area network and hence from there our tool mitm6 could spoof as IPv6 DNS and then serve rouge WPAD data in order to make the host authenticate with the proxy in the network.

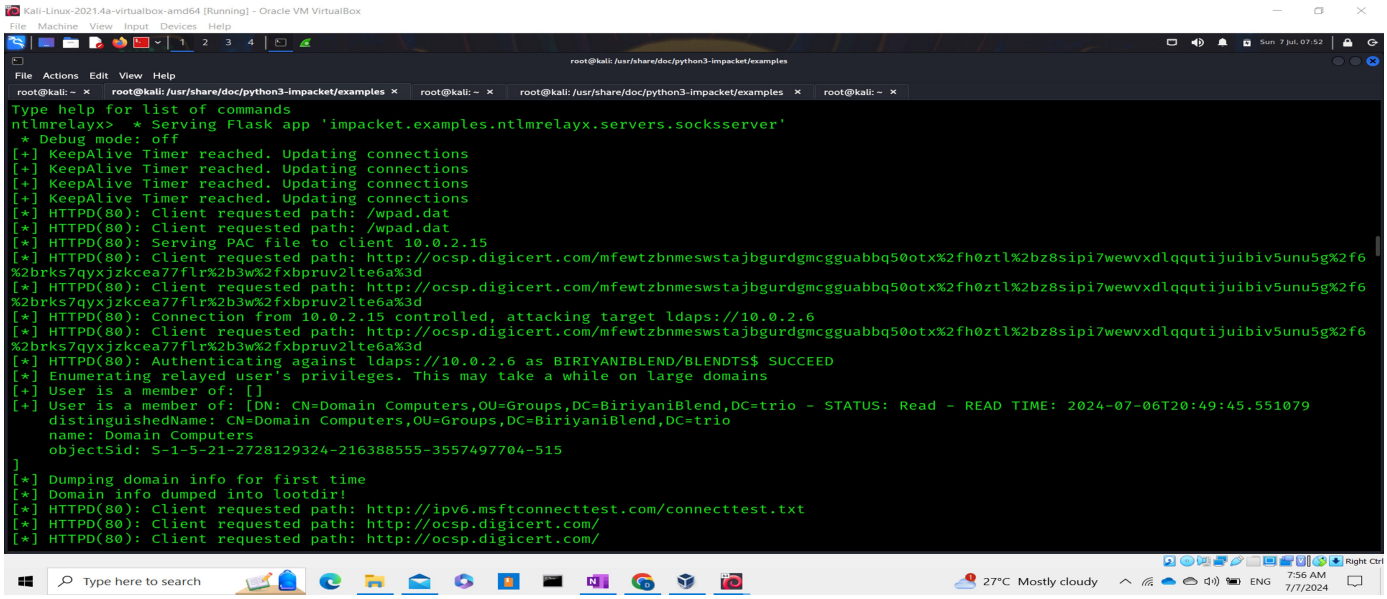
In our case where create two instances where once will reboot the system and login as 'tshaw' and in another instance will reboot the system and login as 'BiriyaniBlend.trio\Administrator' To replicate that in an actual network lot of login and logout activities continues which generates traffic over the network and can be used execute the attack.

Step 3: Rebooting the blendTS PC and logging in as 'tshaw'.

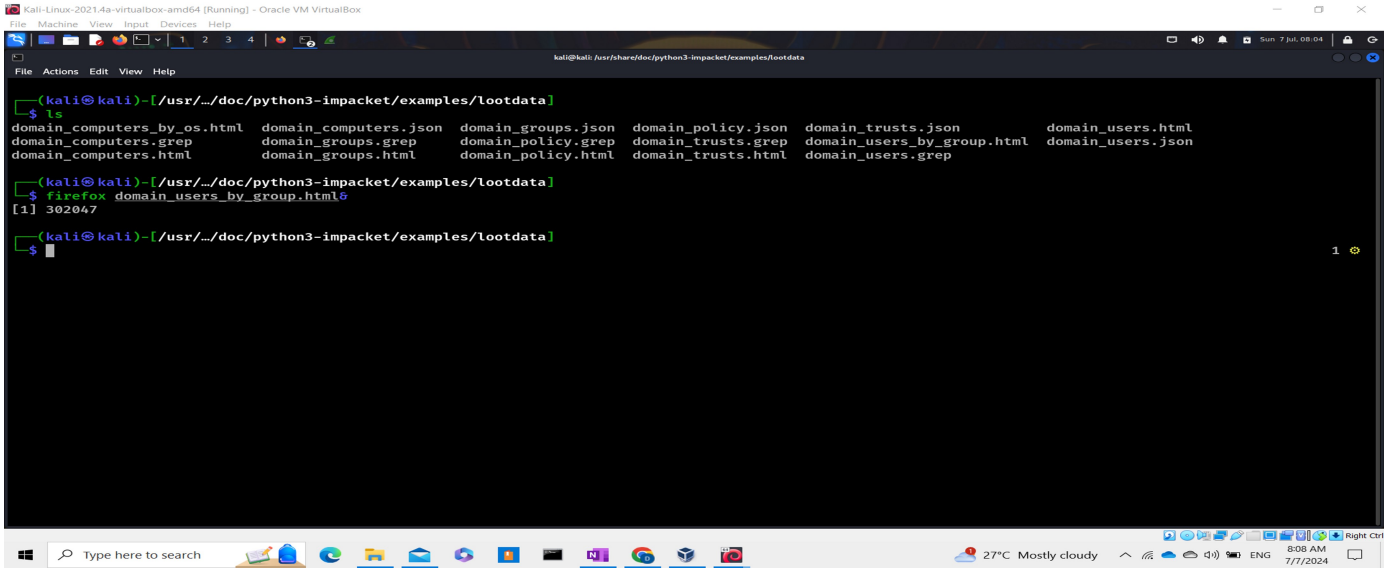


Once logged in, we can observe in the below screenshot that ntlmrelayx.py has relayed the credential to authenticate against ldaps://10.0.2.6 as BIRIYANIBLEND/BLENDTS$ and succeeded in that.

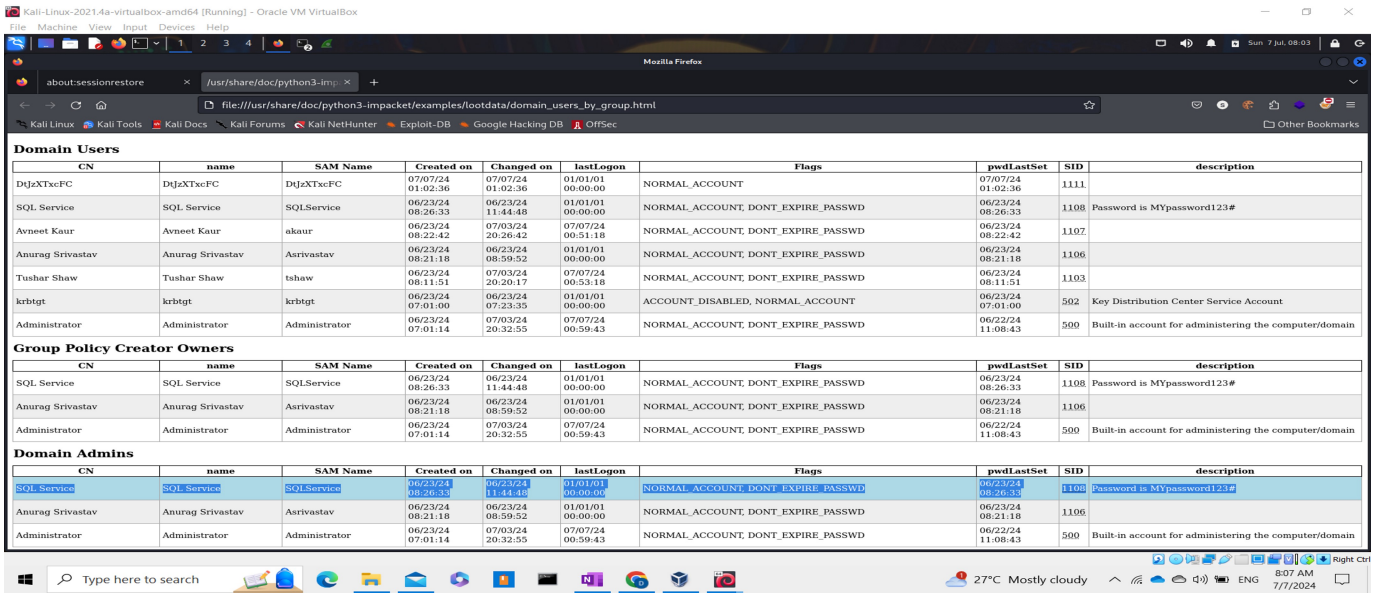
Hence it enumerates the user's privileges and dumps the exfiltrated data accordingly into lootdata file.



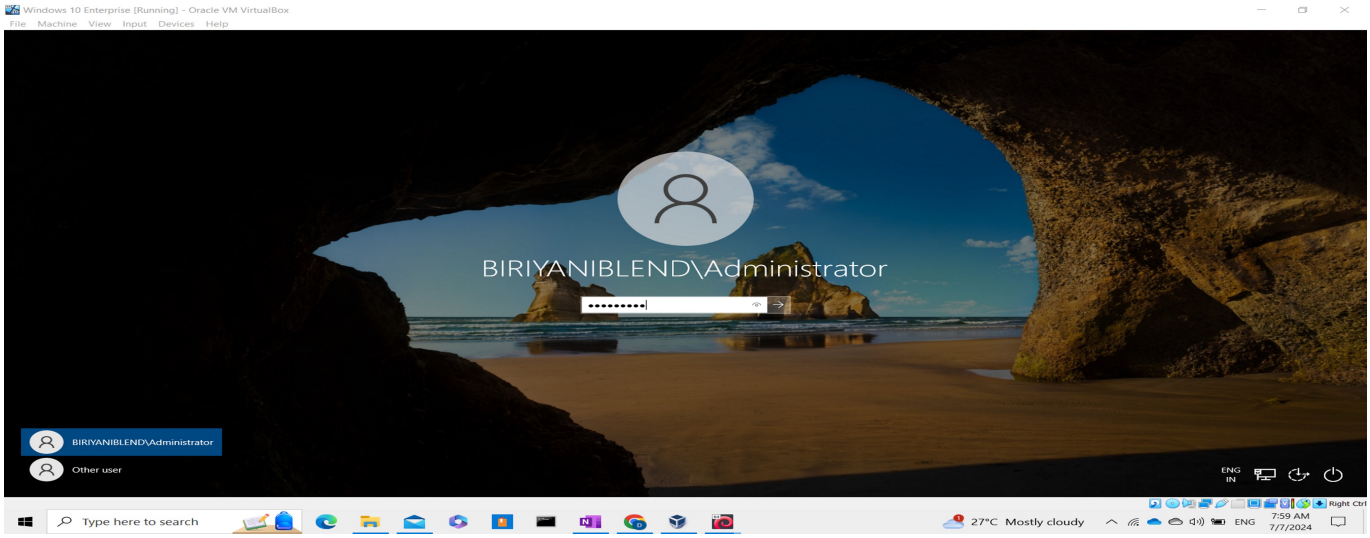
In another terminal window as non-root user, open and observe the lootdata directory.
In the below screenshots we can clearly observe that data related to users, computers, groups, group policy etc has been exfiltrated and hence we can render those html file in order to see it.



Below screenshot show the criticality of the data we scrapped.

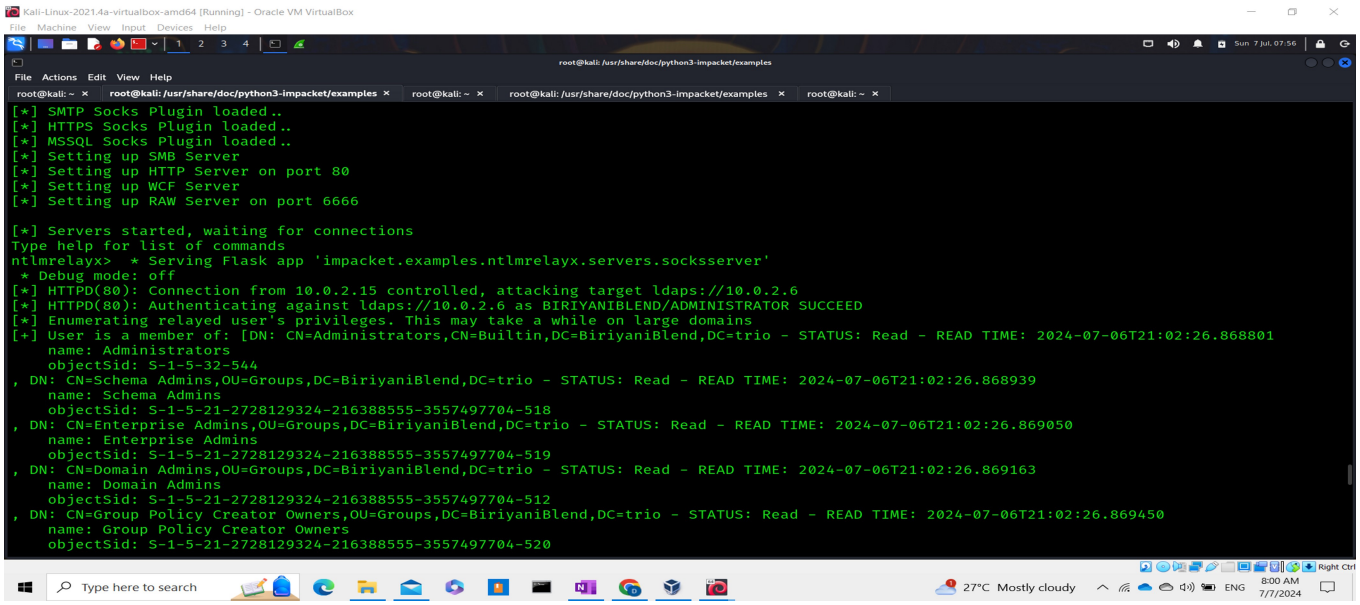


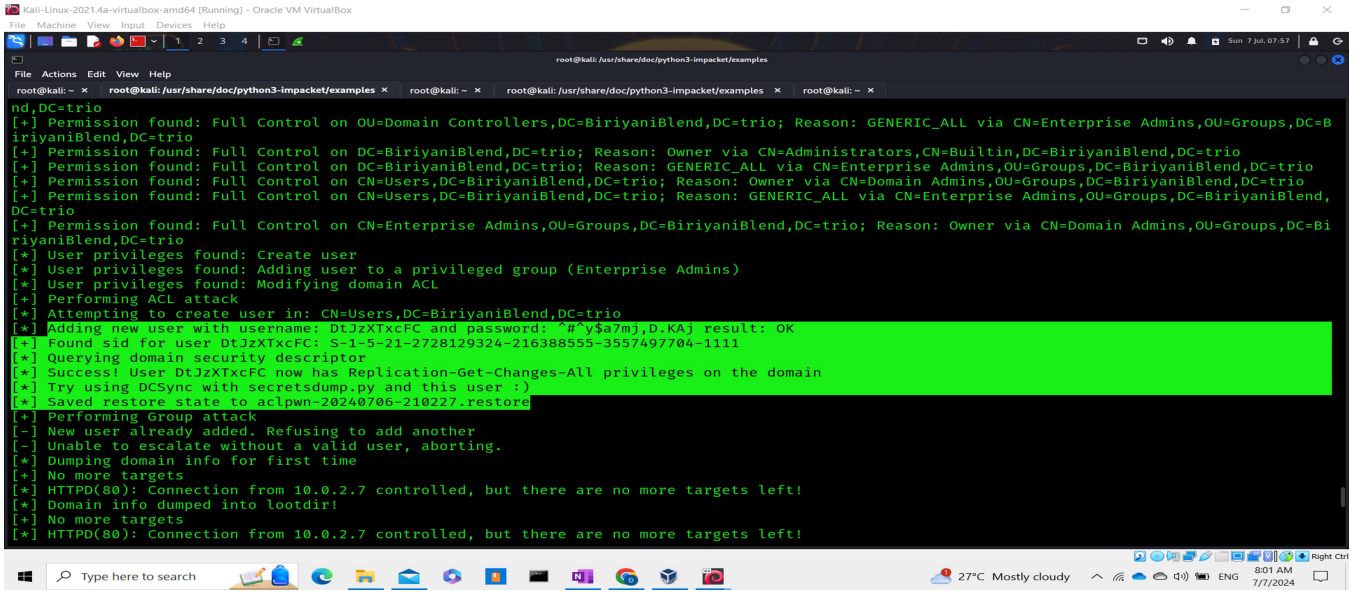
Step 4: Rebooting the blendTS PC and logging in as 'Administrator'.



Once logged in we can clearly verify that ntlmrelayx.py again did the same thing but this time we are administrator and the privileges are of admin hence it moves ahead of just gathering data and also creates a new user along with change in access control list.

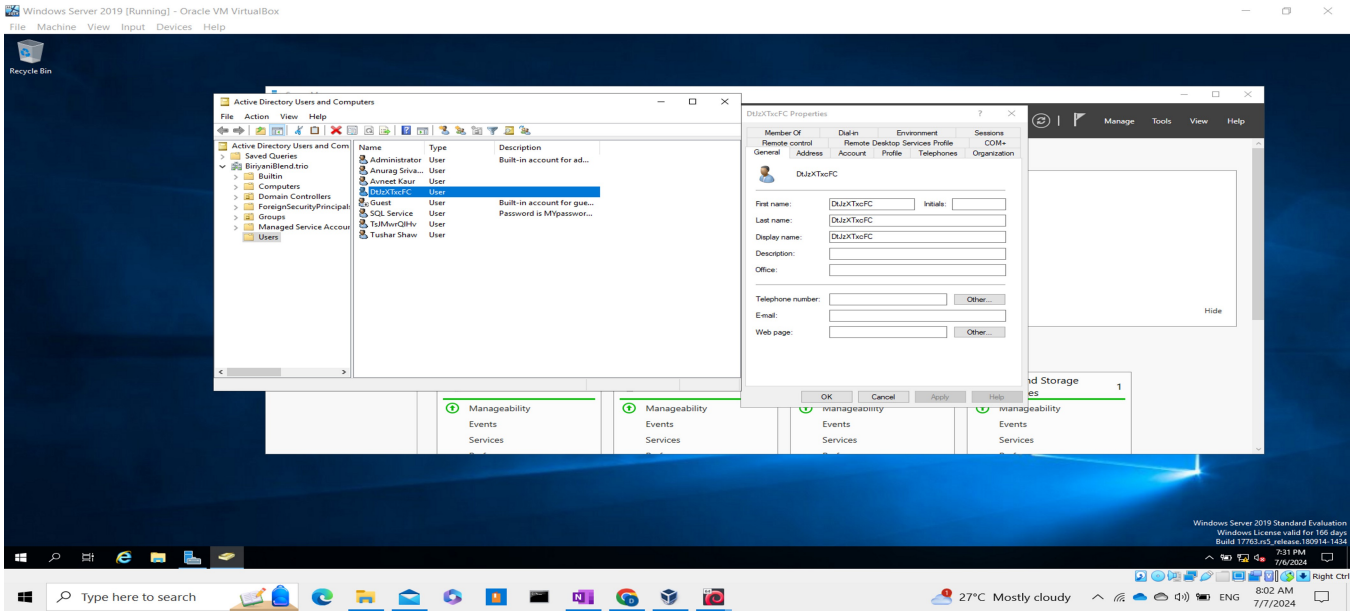
This time the whole domain has been dumped as we can verify from the prompt.





To verify the user created lets examine the Users and Groups at domain controller.

Below screenshot verifies the user has been create.



Mitigation:

1. IPv6 poisoning abuses the fact that Windows queries for an IPv6 address even in IPv4-only environments. If you don't use IPv6 internally the safest way to prevent mitm6 is to block DHCPv6 traffic and incoming router advertisements in Windows Firewall via Group Policy. Disabling IPv6 entirely may have unwanted side effects.

Setting the following predefined rules to Block instead of Allow prevents the attack from working:

- *(Inbound) Core Networking - Dynamic Host Configuration Protocol for IPV6(DHCPV6-In)*
- *(Inbound) Core Networking - Router Advertisement (ICMPv6-In)*
- *(Outbound) Core Networking - Dynamic Host Configuration Protocol for IPV6(DHCPV6-Out)*

2. If WPAD is not in use internally disable it via Group Policy and by disabling the WinHttpAutoProxySvc service.

3. Relaying to LDAP and LDAPS can only be mitigated by enabling both LDAP signing and LDAP channel binding.

4. Consider Administrative users to the Protected Users group or marking them as Account is sensitive and cannot be delegated which will prevent any impersonation of that user via delegation.