NetBios (NBNS) Analysis

NetBios or Network Basic input/output System is the technology responsible for allowing applications on different host to communicate with each others

NetBios is an odder networking "naming and sharing" system that helped computers on a local network find each other and share things like files, printers, and simple services. And mostly used in Windows LANs

What NetBios does:

- Name Service: it let the computers use fixed names instead of IP addresses
- Session Service: set up and manages connections between computers for sharing
- Datagram Service: supports simple messages on the local network

In Current it replaced by: DNS and Active Directory and SMB prefers TCP 445 direct hosting instead of going through NetBios

Wireshark

Global search → nbns

'NBNS'

- Queries : Query details
- Query details could contain "name, Time to live (TTL) and IP address

→ nbns.name contains "E..."

KerBeros Analysis

KerBeros is the default authentication service for Microsoft domains. It is responsible for authenticating service requests between two or more computers over the untrusted network. The ultimate aim is to prove identity securely.

This is used heavily in Windows Active Directory

- Prove who we are without sending our password over the network
- Let us log in once and then access services using short-lived ticket
- Provide mutual authentication

In KerBeros, we have these players:

- Client
- Service center
- KDC (Key-Distribution-Center) → trusted
  - → AS (Authentication Service)
  - → TGS (Ticket Granting Service)

Therefore, KerBeros mostly uses symmetric encryption, and each player has a secret key.

- We will have long term key which is derived from our password
- KDC has its own
- Each service has a secret key known to the KDC and the service

So ~~Kebr~~ Kerberos can encrypt, so that only the intended party can read it, without exchanging password

Actual flow:

1) When we login, our computes asks the KDC's authentication service, saying "I am this, and I want to authenticate"
2) ~~Then, KDC will reply:~~
   This request usually includes pre-auth (a timestamp, encrypted with our password-derived key, to prove we know the password without ~~knowing~~ sending it)
3) Then KDC creates a TGT and encrypts it with a secret key that only the KDC and TGS can decrypt
4) We store that TGT and later present it back to request service tickets. Which means we can't open the TGT.
5) KDC also sends us the session key, which is encrypted with our long-term key
6) Then we use that session key to ~~decrypt~~, encrypt and authenticate our next request to the TGS
7) The KDC/TGS decrypts the TGT and verifies us, and then issues a service ticket

Kerberos commonly uses time-based authentication to prevent replay attacks
- clock sync matters

Kerberos typically uses port 88 (UDP/TCP) in Active Directory environments

If it fails, we may fall back to NTLM in some setup.

Wireshark

Global search → Kerberos

User account search: → ~~kerberos.CNameString~~
- Cnamestring: The username
  - kerberos.CNameString contains 'keyword'
  - kerberos.CNameString and !(kerberos.CNameString contain "$")

"Kerberos" option: ~~Packet Grabbing~~

- pvno: Protocol version
- realm: Domain name for the generated ticket
- sname: Service and domain name
- address: Client IP address and NetBios name

- kerberos.pvno == 5
- kerberos.realm contains ".org"
- kerberos.SNameString == "krbtg"

Questions

What is the MAC address of the host "Galaxy A30"

dhcp.option.hostname contains "Galaxy"

How many NetBios registration requests does the "LIVALJM" workstation have?

(nbns.name.contains "LIVALJM") && (nbns.flags.opcode == 5)

Indicates only registrations

Which host requested the IP address "172.16.13.85"

(dhcp.option.dhcp == 3) && (dhcp.option.requested_ip_address == 172.16.13.85)

request

What is the IP address of the user "u5"

kerberos.CNameString == "u5"

What is the hostname of the available host in the Kerberos packet

kerberos.CNameString contains "$"

available