Disabling NTLM Authentication Guide – part 6 – RDP

willssysadmintechblog.wordpress.com/2023/09/05/disabling-ntlm-authentication-guide-part-6-rdp

September 5, 2023

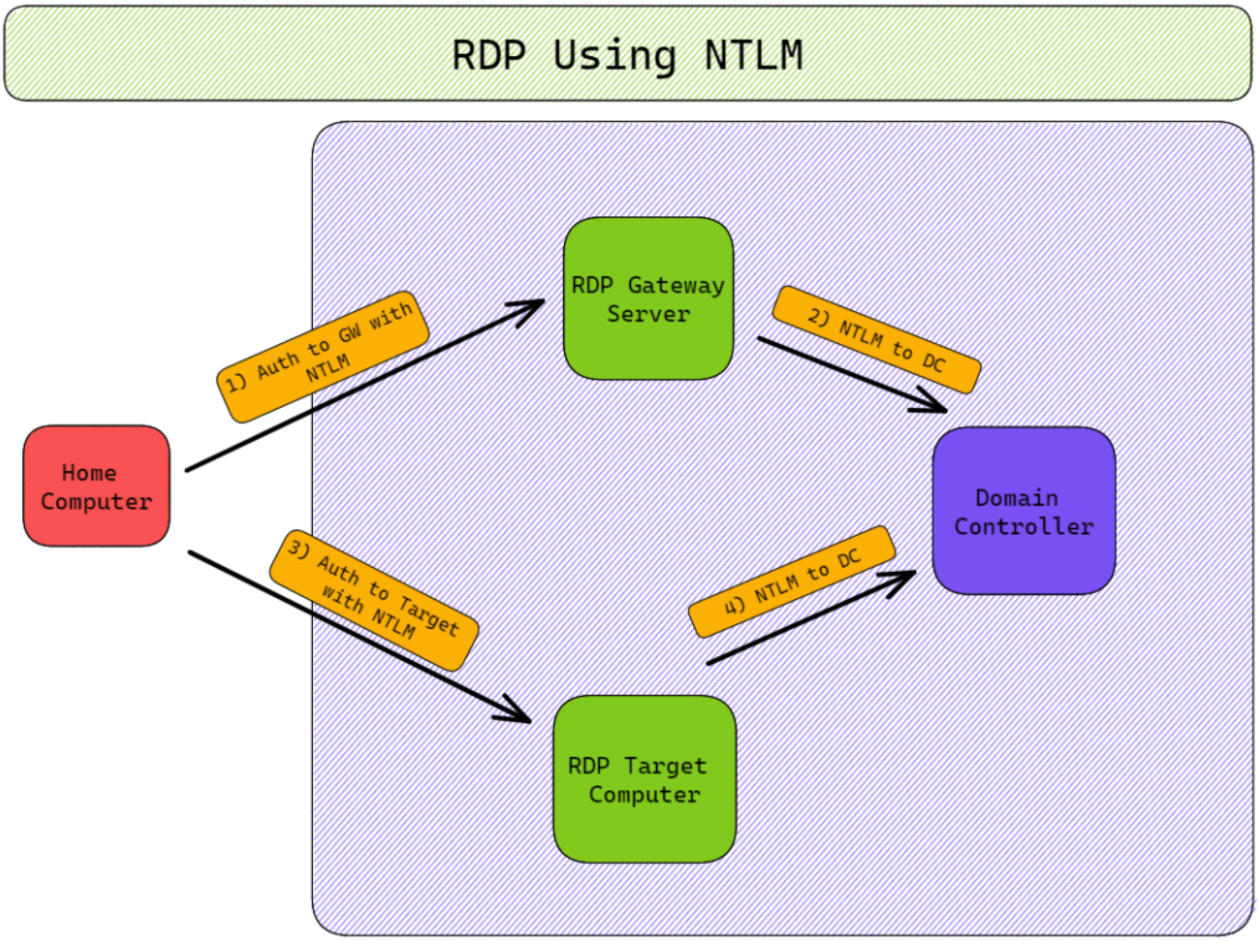
Part 5: Disabling NTLM Authentication Guide – part 5 – Printers and Scanners

Part 7: Disabling NTLM Authentication Guide – part 7 – Kerberos Logs

Background

If your organization utilizes Remote Desktop Protocol (RDP) you're going to run into this one. Getting RDP to work in an environment with NTLM disabled, and validating that it would work in our production environment, took weeks. Using RDP without NTLM is not well documented on the internet.

RDP presents unique challenges to a post-NTLM environment. RDP use means that users authenticate to Active Directory (and varied endpoints on campus) from the internet, often with personal computers without VPN access. Doing Kerberos requires access to the Kerberos servers, running on port 88 on domain controllers, but these ports are restricted to on-campus use. The typical path of authentication using NTLM for RDP is as follows. The arrows don't represent the flow of network traffic, but a simplified flow of where authentication requests are happening.



RDP Gateway

The RDP Gateway server (GW) acts as a proxy to allow RDP connections to get in from the internet. Users must authenticate to use the GW, which uses NTLM. The GW cannot be migrated off NTLM and needs an exception to continue using it, per Microsoft.

RDP Server (Target Computer)

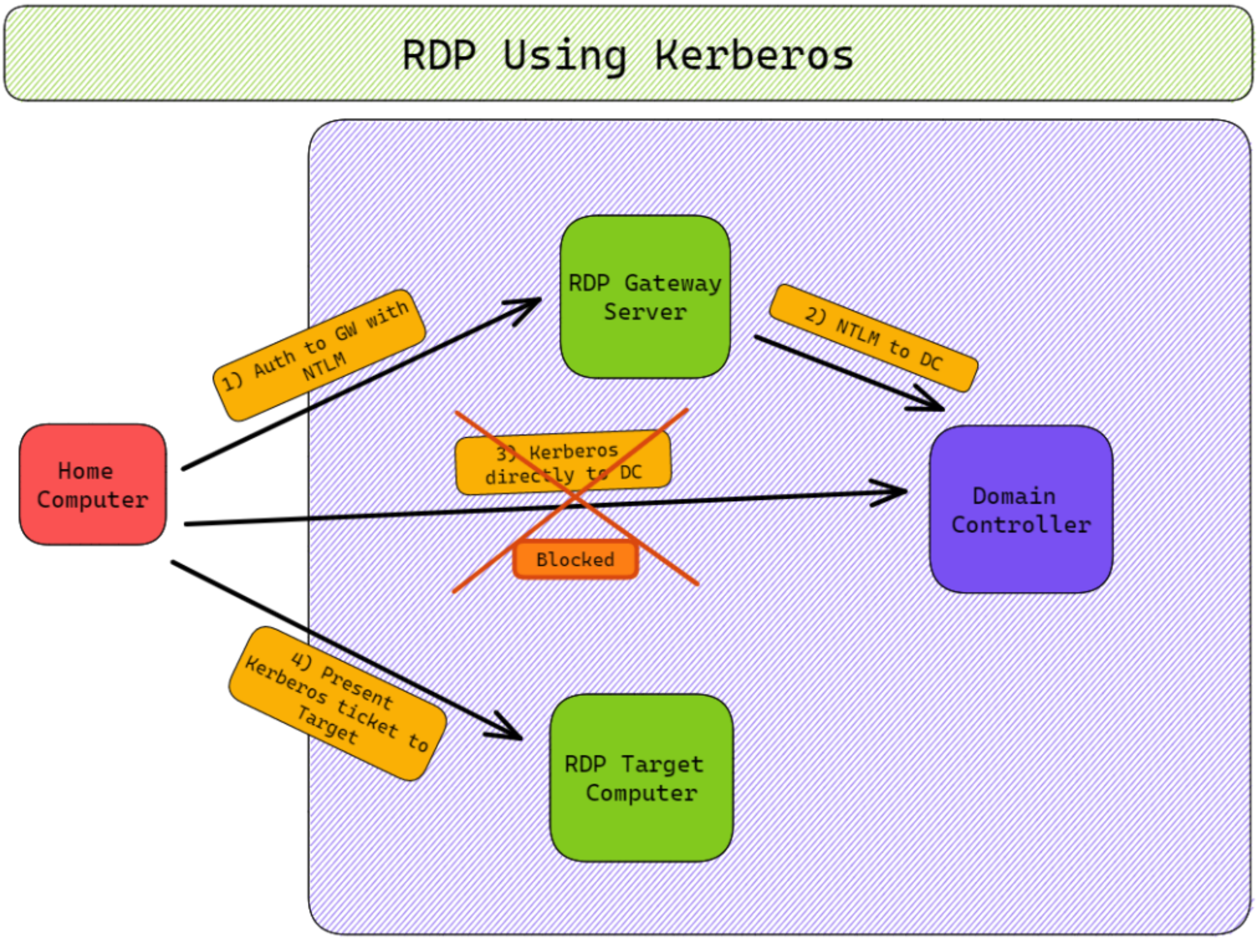
Once a user is authenticated to the gateway, they then attempt to authenticate to the RDP server, or target workstation they are connecting to. This can be done through NTLM or Kerberos.

Network Level Authentication Issues

An RDP security feature called Network Level Authentication (NLA) requires that you authenticate successfully to the target computer before actually getting an RDP session. If you had an RDP session before you were authenticated, you could enter login credentials on the target computer's login screen over RDP, but NLA prevents this.

Privacy Settings

When NTLM is disabled, this authentication to the target computer must be done over Kerberos. Unfortunately, this isn't possible from the internet as the domain controllers are not accessible from the internet. You get this scenario:



Proposed Solutions

The following solutions were proposed to fix this issue:

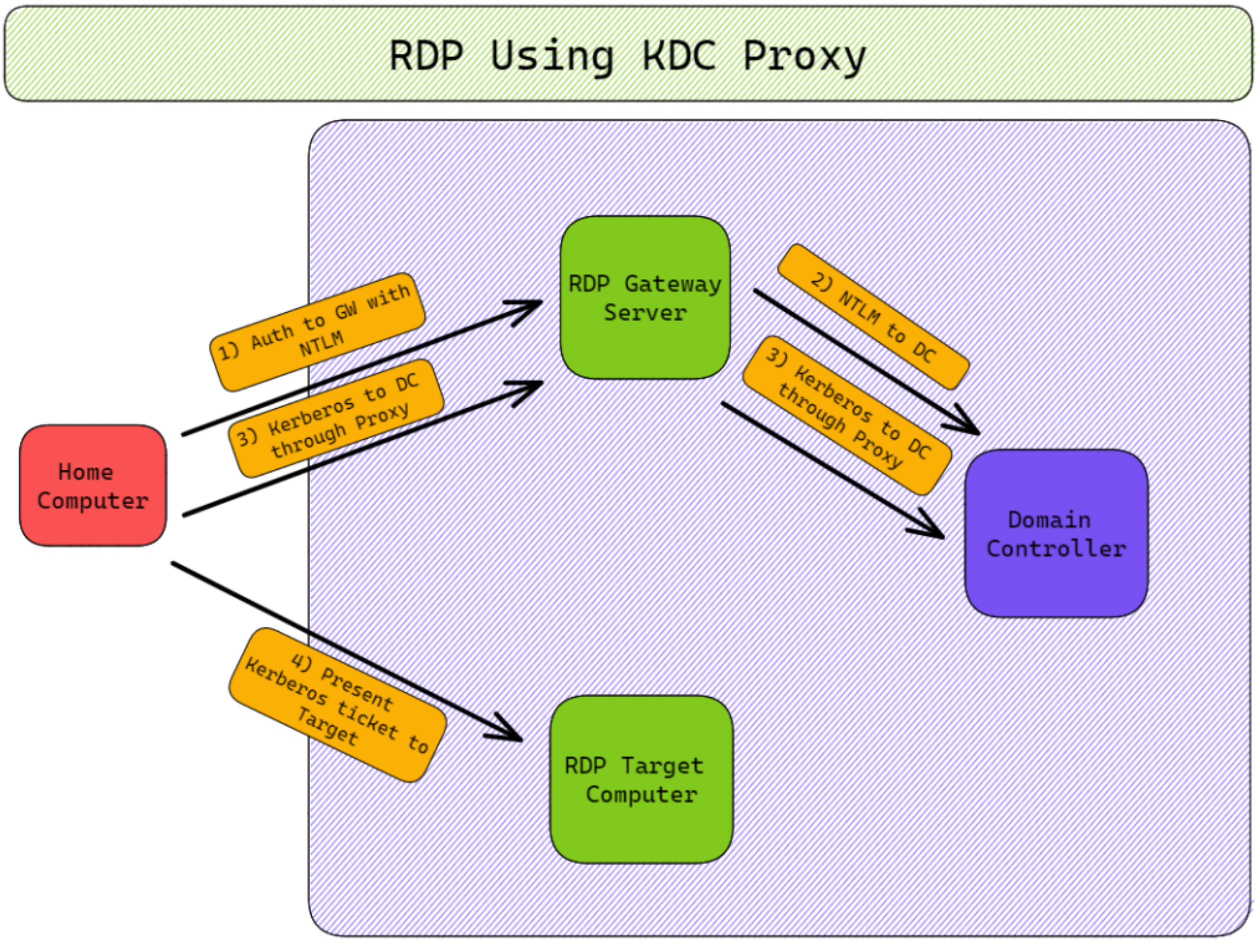
- Give all workstations NTLM exceptions
    - Allows NTLM on potentially thousands of computers, the very thing we’re trying to avoid
    - Potentially requires all of these computers to be entered into a text field that domain admins must maintain in the DC NTLM exception list
- Allow access from internet to domain controllers on port 88 to allow Kerberos to work from home
    - Open up DCs to internet, which isn’t desirable in case there are Kerberos server vulnerabilities. DCs are too important to expose directly to internet
    - Wasn’t super reliable in our test envoronment
- Use a VPN before connecting to RDP, to get you behind the border firewall

    Many RDP users use personal computers, which you may not want running VPNs
- Use KDC Proxy (more on this below)

    Not all clients support this

KDC Proxy

RDP GWs have a built-in feature called the Key Distribution Center Proxy (KDC Proxy). This allows the GW to act as a proxy between your client and the domain controllers, passing your Kerberos traffic over HTTPS. With this feature you don’t need direct access to domain controllers to do Kerberos, so Kerberos can be done from the internet!



Clients that support KDC Proxy use it by default if available. Most of RDP authentication today already uses KDC Proxy already, instead of vanilla NTLM like described above.

You can see if a KDC Proxy is processing authentication requests by viewing this event log on the GW server: *Microsoft-Windows-Kerberos-KdcProxy/Operational.* When you try connecting to the GW and the proxy is working, you'll see activity in the log with EventIDs 309 and 400.

Issues

When setting up GWs in my test environment, the KDC Proxy would not always work. This lead me to write-off the feature as broken early on in my testing. I don't know why the KDC Proxy didn't work at first, but it started working later. I could see the point in the event logs where the Proxy started creating events, and I had been testing the GW for long before when it started. Since then, GW servers I've created have had their KDC Proxies working right away.

Forget about setting KDC Proxy settings on clients via GPOs, Registry, or RDP configuration files, that never made a different when I was having issues with it. A client will auto-detect if a KDC Proxy is present on a GW server and use it. I also never found benefit with changing settings on the GW. If it was going to work it worked automatically, and there was nothing I could do to make it work when it didn't want to work. I went through all the articles I could find out there to get the KDC Proxy working, but nothing fixed it. Finally, after setting up 2-3 gateway servers, the KDC Proxy started working out of nowhere. Ever since then it has worked flawlessly. It's my opinion that the KDC Proxy is in dire need of good documentation from Microsoft! Something more than blog posts from Microsoft employees is needed here. Those are helpful, but let's make it official, Microsoft.

Client Issues

Unfortunately, not all RDP clients support this awesome KDC Proxy feature. Clients that I know do support it:

> mstsc.exe on Windows (built-in RDP client)

Clients that don't support it:

- Microsoft Remote Desktop from the Windows Store
- Microsoft Remote Desktop from the Mac App Store
- Remmina on Linux

If you attempt to connect to RDP using a KDC Proxy you get generic authentication errors or Cred SSP errors on your client.

As far as I know this lack of feature on some clients is entirely undocumented! It was only through lots of testing that I figured this stuff out. If someone knows how to get these clients working with a KDC Proxy, please let me know!

Privacy Settings
Client Solutions

Your organization will need to figure out how to mitigate the RDP client shortcomings mentioned above. We chose a varied approach, based on a couple factors:

- Is the RDP client computer a personal computer, or organization owned computer?
- Are they connecting from a Windows computer or something else?

Based on those answers, we steered clients toward one of the following:

- Connect to on-premises via VPN before doing remote desktop
- Disable NLA on their target computer. You want to be careful where RDP (port 3389) is exposed on the hosts firewall, and especially so with NLA disabled. This solution also requires users to enter their login credentials twice when connecting to RDP
- Use a different RDP client (mstsc.exe)

Part 1: Disabling NTLM Authentication Guide – part 1 – Prerequisites

Part 2: Disabling NTLM Authentication Guide – part 2 – Logs

Part 3: Disabling NTLM Authentication Guide – part 3 – Migrating to Kerberos

Part 4: Disabling NTLM Authentication Guide – part 4 – NTLM Restrictions and Testing

Part 5: Disabling NTLM Authentication Guide – part 5 – Printers and Scanners

Part 6: Disabling NTLM Authentication Guide – part 6 – RDP

Part 7: Disabling NTLM Authentication Guide – part 7 – Kerberos Logs