[Backup] How did I design and build a complex AD set

Hi Folks, today I would like to share how did I design and build a vulnerable AD set. Before moving to this topic, let me introduce the motivation and some features of this AD set.

MOTIVATION

I know there are few scripts can automate the process of generating common AD misconfigurations such as DACL abuse, weak credential, kerberoasting, etc. If you are interested in them, here are the github repo: https://github.com/WaterExecution/vulnerable-AD-plus and https://github.com/Orange-Cyberdefense/GOAD. These authors already did a great job, they make the process simple and fast. However, some other common elements in AD exploitation cannot be produced easily only with script, so some manual configuration and setup is also very important. Besides, I do not want my vulnerable AD set to be a purely AD exploitation. I hope it is more complex, difficult, and realistic.

FEATURES

1: It is not CTF style, no side quest. All flags are on Linux home folder or Windows Desktop. Its style is similar to many famous AD labs like CPTX, Cybernetics, CRTP, etc.

2: The AD consists of 6 machines, including 2 Linux domain joint machines. Many people are already familiar with AD exploitation in Windows environment, but how about Linux domain joint machines? You even need to exploit the AD from your Kali VM.

3: Multiple services and apps make the vulnerable AD more fun and complex, such as FTP, SMTP, POP3, IMAP, Samba, ElasticSearch, WordPress, Kibana, etc.

4: Few rabbit holes, but not just for misleading you. They are reasonable. Get RCE from a web app? But it will not help too much. A lot of privilege escalation vectors? But they are not necessary.

5: Basic OSINT and inference according to context.

6: Hardened machines. They implemented latest Windows Defender, AppLocker, etc. But I don’t think they will be the biggest issue, enumeration does matter.

7: Classic elements in AD: SQL Linked Server, Kerberos Delegation, Kerberoasting/ASREPRoasting, Credential Reuse

8: Some barriers during typical exploitation. Copy and paste steps in an AD exploitation cheat sheet? They will not work, you need to understand why your exploitation failed.

PREPARATION

During the design, I downloaded multiple apps/tools, and referred many articles. But before building the AD set, only 2 things are required.

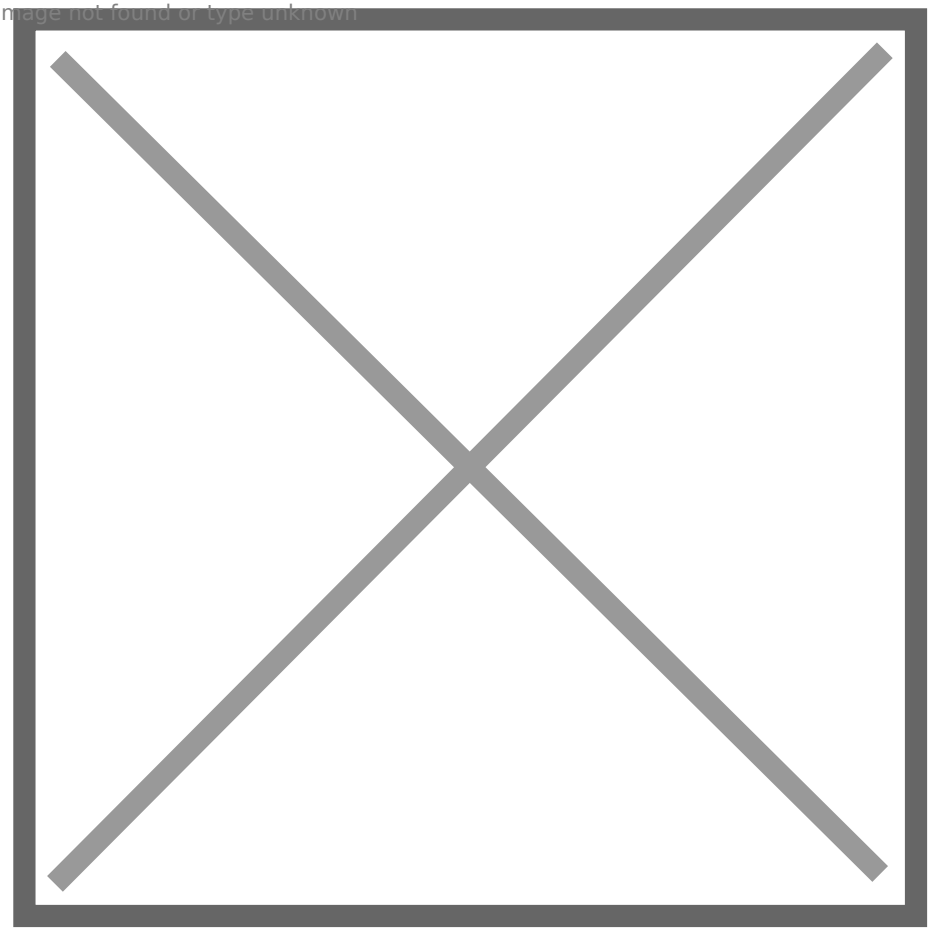
Windows Server 2019: https://www.microsoft.com/en-us/evalcenter/evaluate-windows-server-2019

Windows 10: https://www.microsoft.com/en-us/software-download/windows10%20 (You can also use Windows Server 2019 instead)

Ubuntu 22.04: https://ubuntu.com/download/desktop

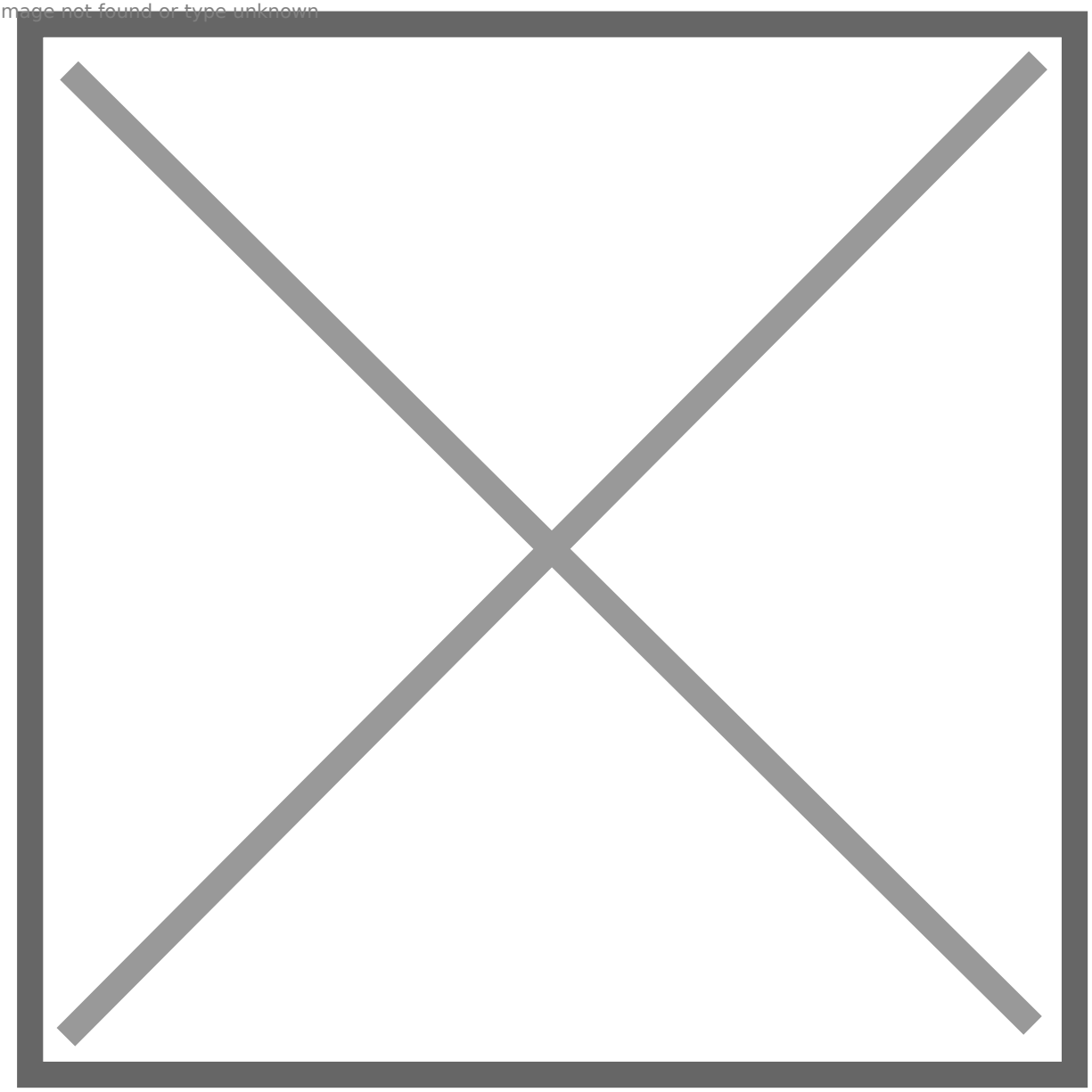
I used VMWare workstation to host these VMs, and I used Bridged Network. I tested NAT network, it also works well! After creating a Windows Server 2019 VM, do not forget to uncheck **Connect at Power Up** (in screenshot), in section **floppy disk**, otherwise you cannot install the OS successfully.

Image not found or type unknown



How to assign hardware resource to these VM? I list them on the following table. In my opinion, they are all above the required resources, I feel each VM runs smoothly.

Image not found or type unknown



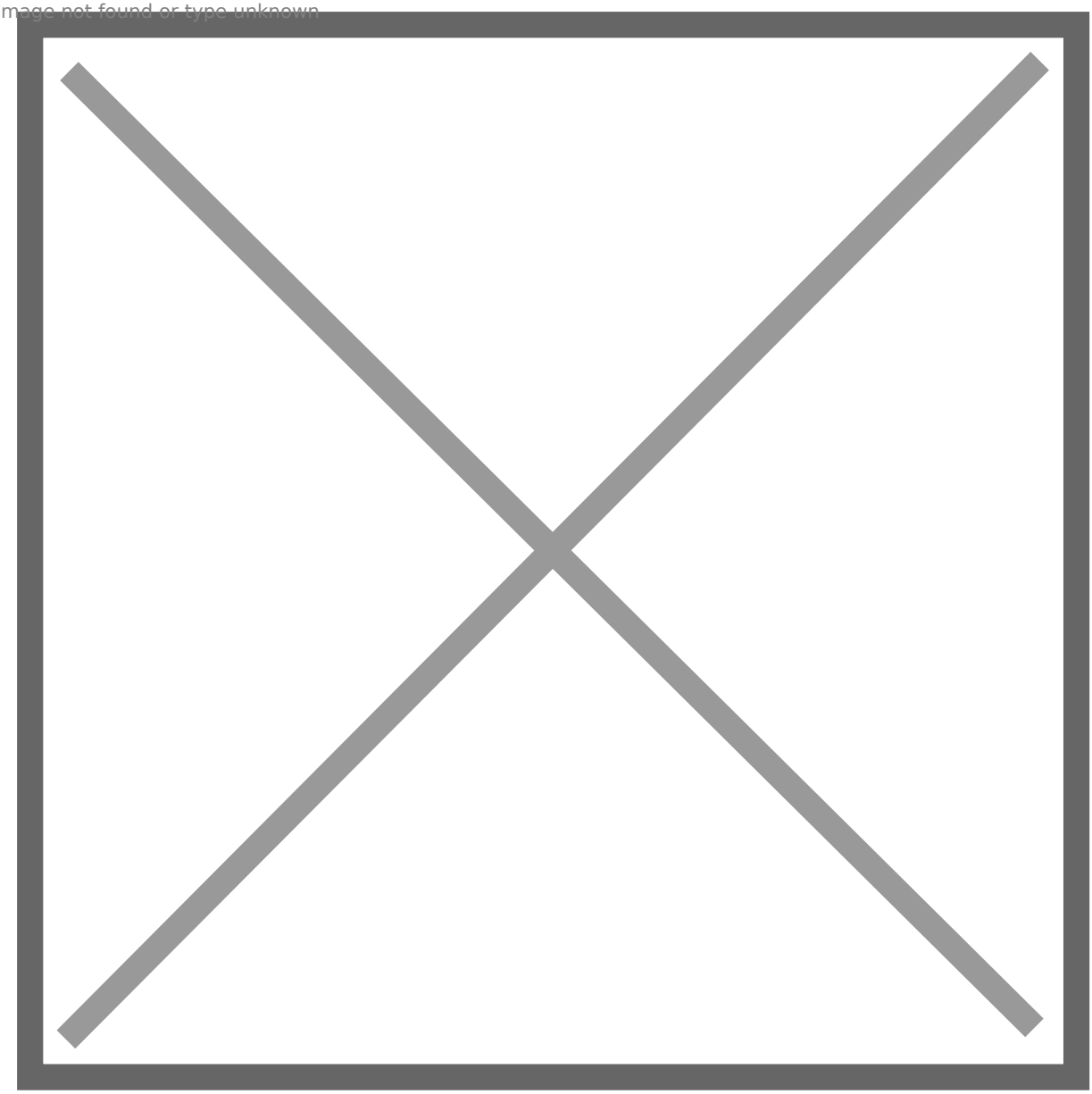
Forget to mention that, I used a Windows 10 pro as the client server in domain. I cannot remember clearly where did I download the image. If it is not convenient for you to download a Windows 10 pro image, you can absolutely use Windows Server 2019 instead, it does not matter. After installing all VM, we can start to configure the OS.

CONFIGURATIONS AND DESIGN

Just clarify, this part is not a detailed guidance for building an AD environment. Instead, this part focuses more on the design. Of course, I will absolutely go through some technique difficulties and how did I resolved them.

Let’s take a look at all machines and their roles.

Image not found or type unknown



Web01 simulates a public-facing server in the domain, external user has access to its services. It hosts multiple services, including web apps, SMB, SMTP, POP3, etc.

File01 simulates an internal file server in the domain, because it is running a FTP server. Domain user can exchange file on this host.

Client01 simulates a client computer in the domain. Domain user Helen Park is the owner of it. Helen is a member of Help Desk group, so she has some permissions.

SRV01 simulates a normal server in the domain, it has an SQL instance. It is also linked to an SQL instance on SRV02

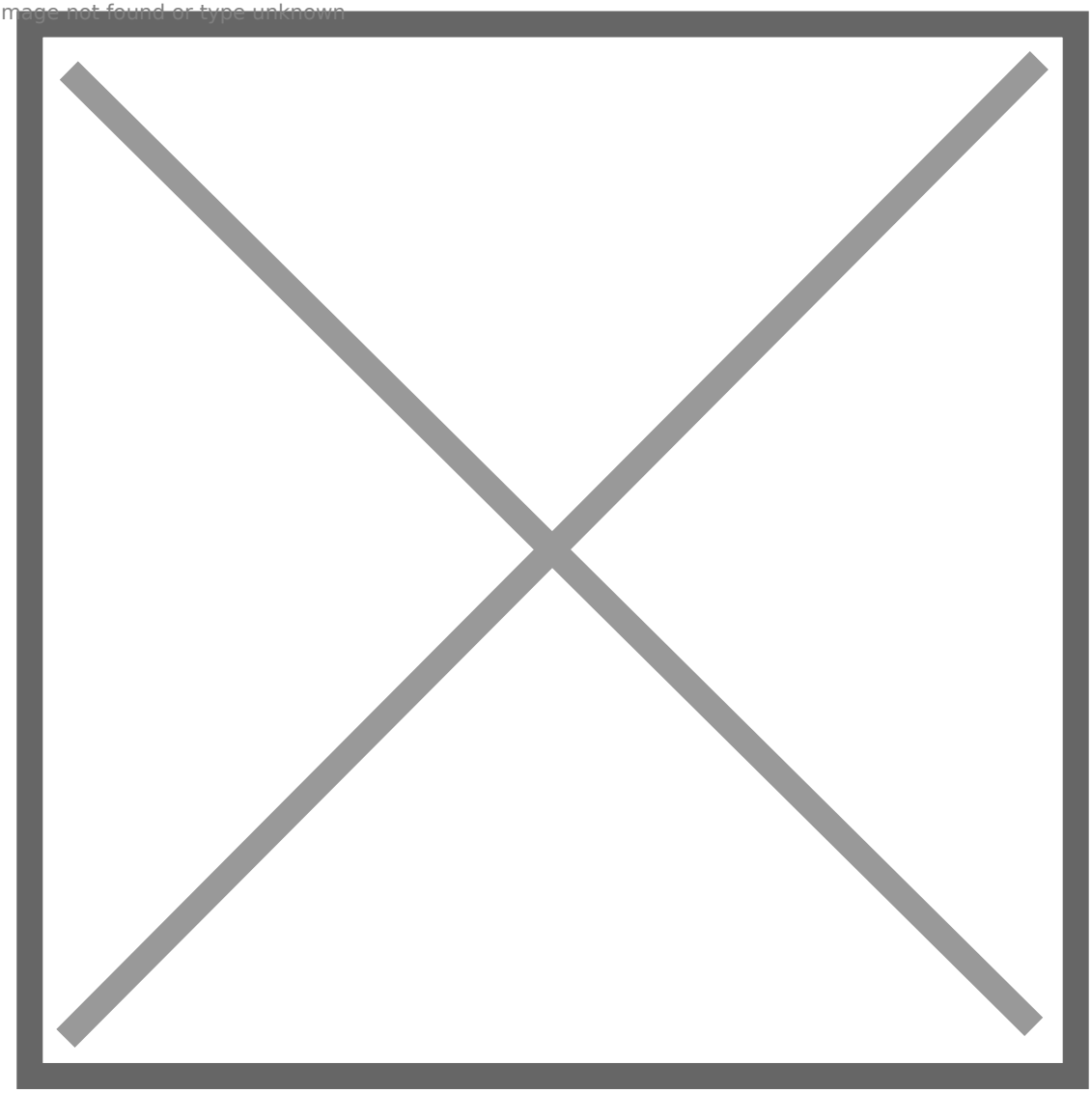
SRV02 simulates another server in the domain, it not only has an SQL instance, but also is able to delegate other domain users, except those protected high-privileged ones.

DOMAIN CONTROLLER

dc.blackops.local

First, we need to configure the domain controller. There are already many articles about it, so I recommend you to check this article: https://kamran-bilgrami.medium.com/ethical-hacking-lessons-building-free-active-directory-lab-in-azure-6c67a7eddd7f. You can jump to **[Configuring Services]** and continue.

Image not found or type unknown



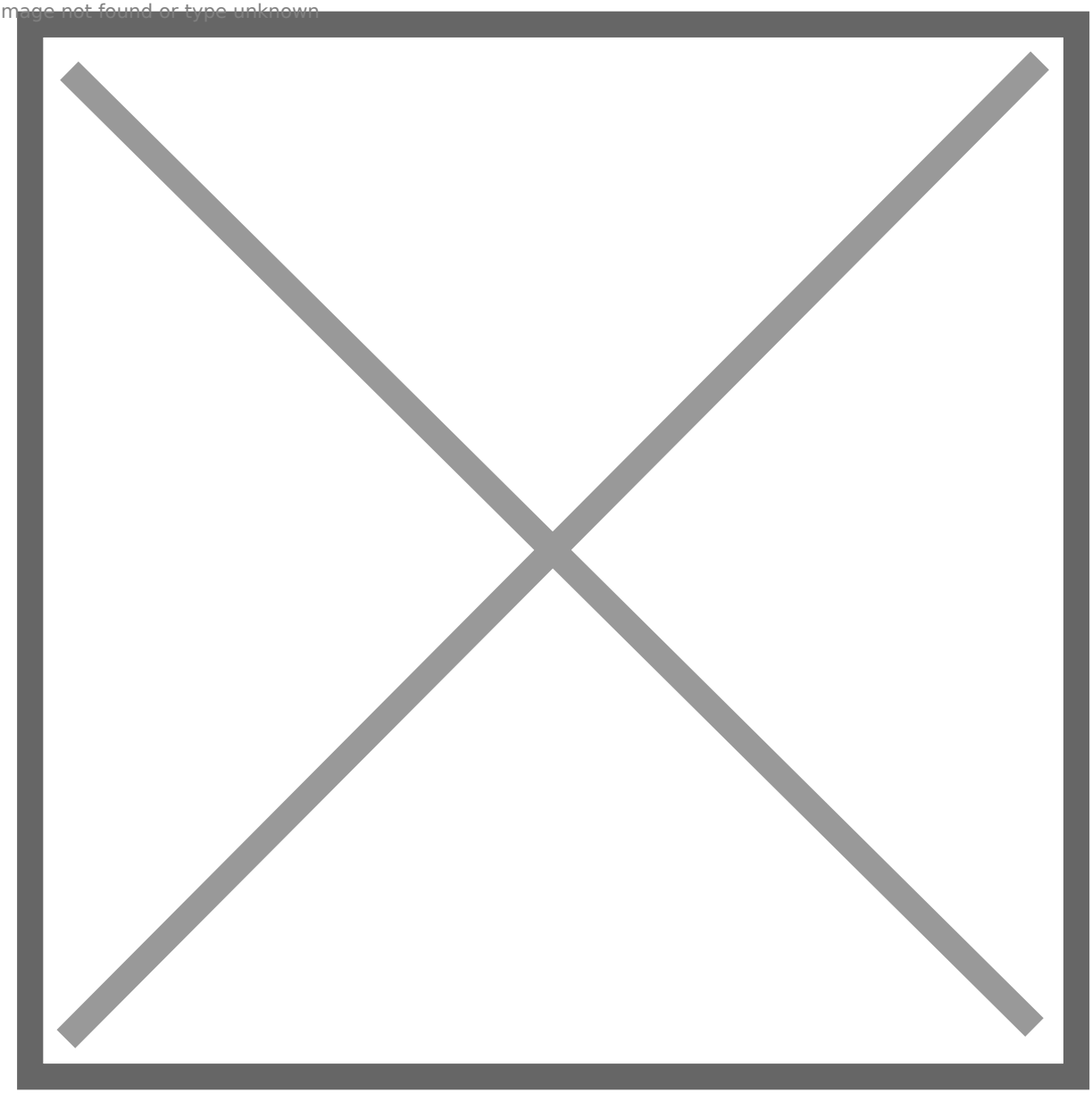
If you just want to replicate my AD set, you can stop at [Configuring Certificate Services] since I did not adopt ADCS this time. If you are interested in this part, you can absolutely continue to read. And I plan to add ADCS feature to my next AD set.

Personally I set the domain as **blackops.local**, the NETBIOS name is **BLACKOPS**, and IP for domain controller is **192.168.0.56**. Then open Active Directory Users and Computers application,

let’s make some changes.

First, let’s create some domain users. Of course, you can create more domain users to increase the enumeration difficulty.

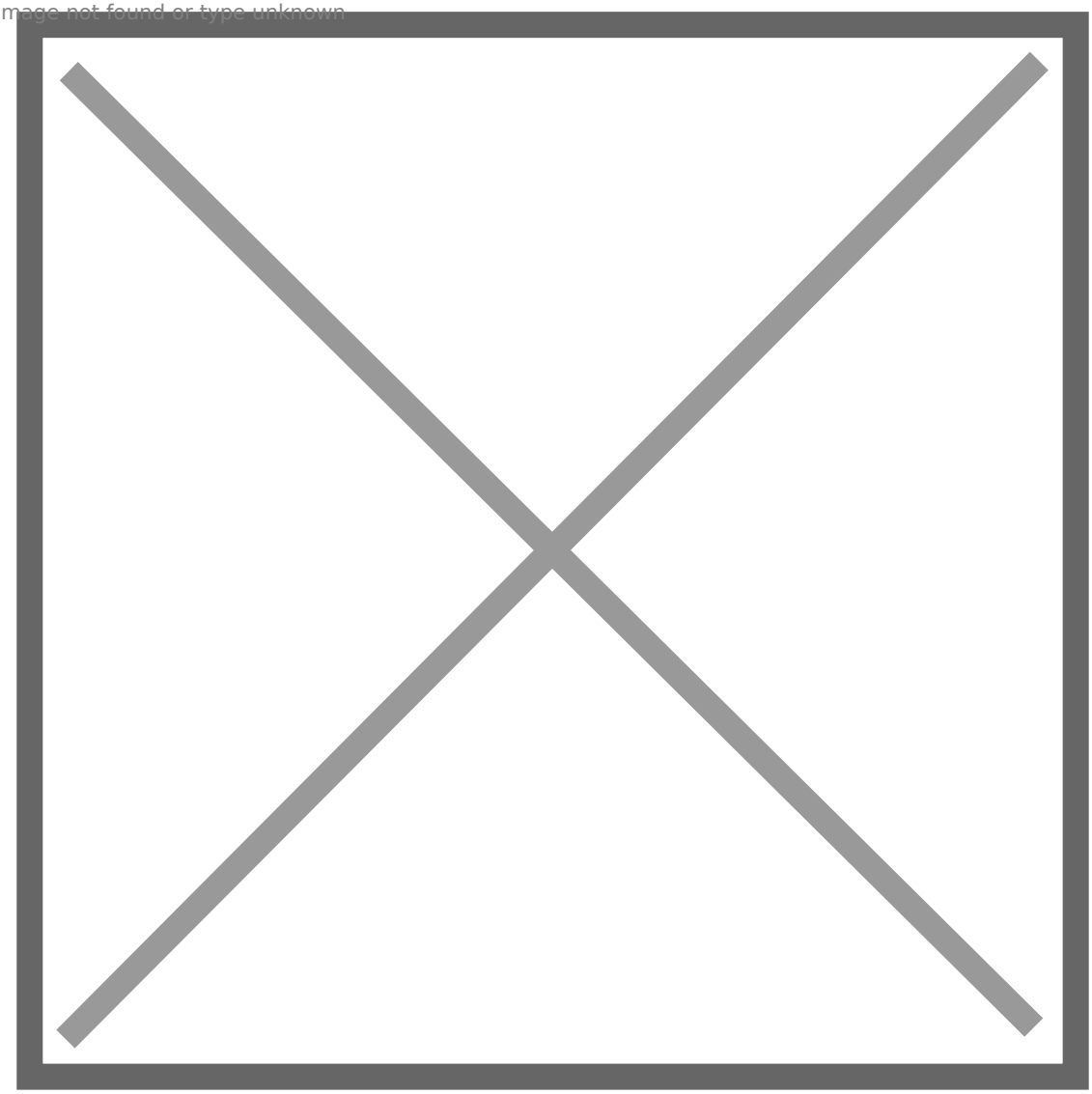
Image not found or type unknown



As you notice, there are some accounts with weak password. I set those weak passwords on purpose. **ir_operator** and **df_operator** share the same password to make room for credential use. In exploitation chain, ir_operator can be set a SPN, then ir_operator can be kerberoasted, this is the reason why I set a weak password for both of them. As to svc_sq1 account, you may find that this is a honeypot account, because it is sq1 not sql : D You can easily kerberoasting svc_sq1 and crack the password, but it will not help at all. In reality, your attack will be logged then blue team will notice it. Anyway, since there are few weak passwords, we must eliminate dictionary attack and brute-force attack, so we need to implement account lockout policy. This article tells you how to achieve this: https://www.windows-active-directory.com/account-lockout-policy-active-directory.html#:~:text=Double%2Dclick%20the%20domain%20to,Policies%20%E2%86%92%20Ac

count%20Lockout%20Policy.

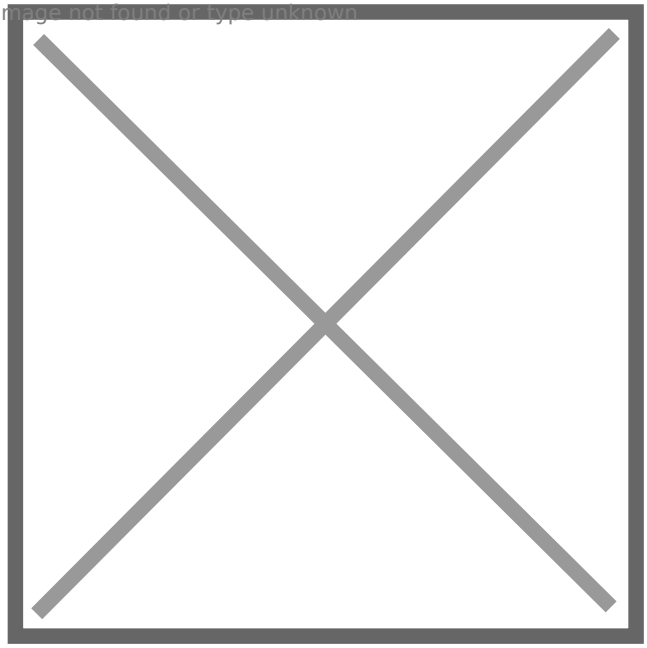
Image not found or type unknown



Apart from few passwords, I specified few passwords such as russell.adler’s, these passwords cannot be cracked with a normal dictionary, but they will be used later in design steps, you can change them but just change them in following steps as well.

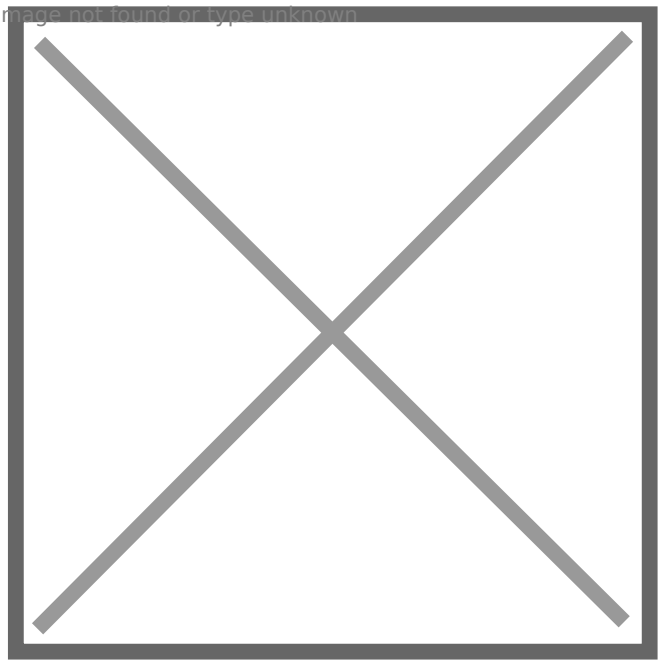
By the way, I set Administrator as a protected user, which cannot be delegated. It not only makes the environment more realistic but also increases the difficulty when abusing delegation.

Image not found or type unknown



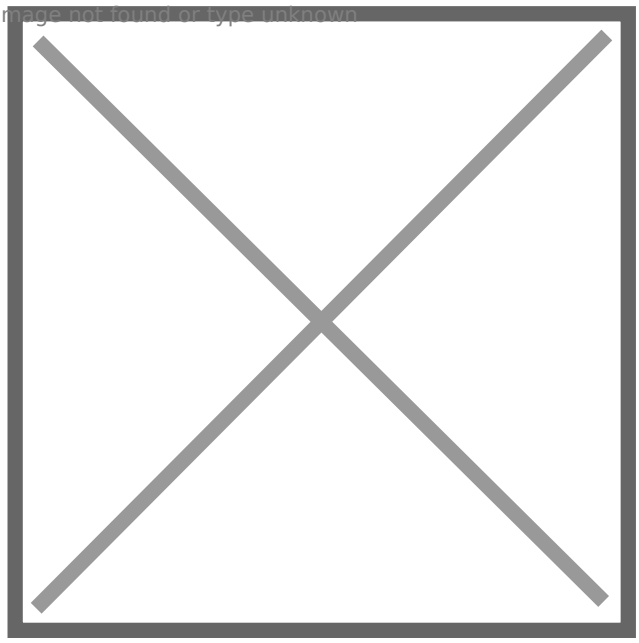
I added an OU called Service Accounts, and I moved svc_sql and svc_sq1 to this OU. Since they are designed as service accounts, we need to set SPN for them. svc_sq1 is a honeypot account, so it is easy to set, as long as the SPN is in correct format.

Image not found or type unknown



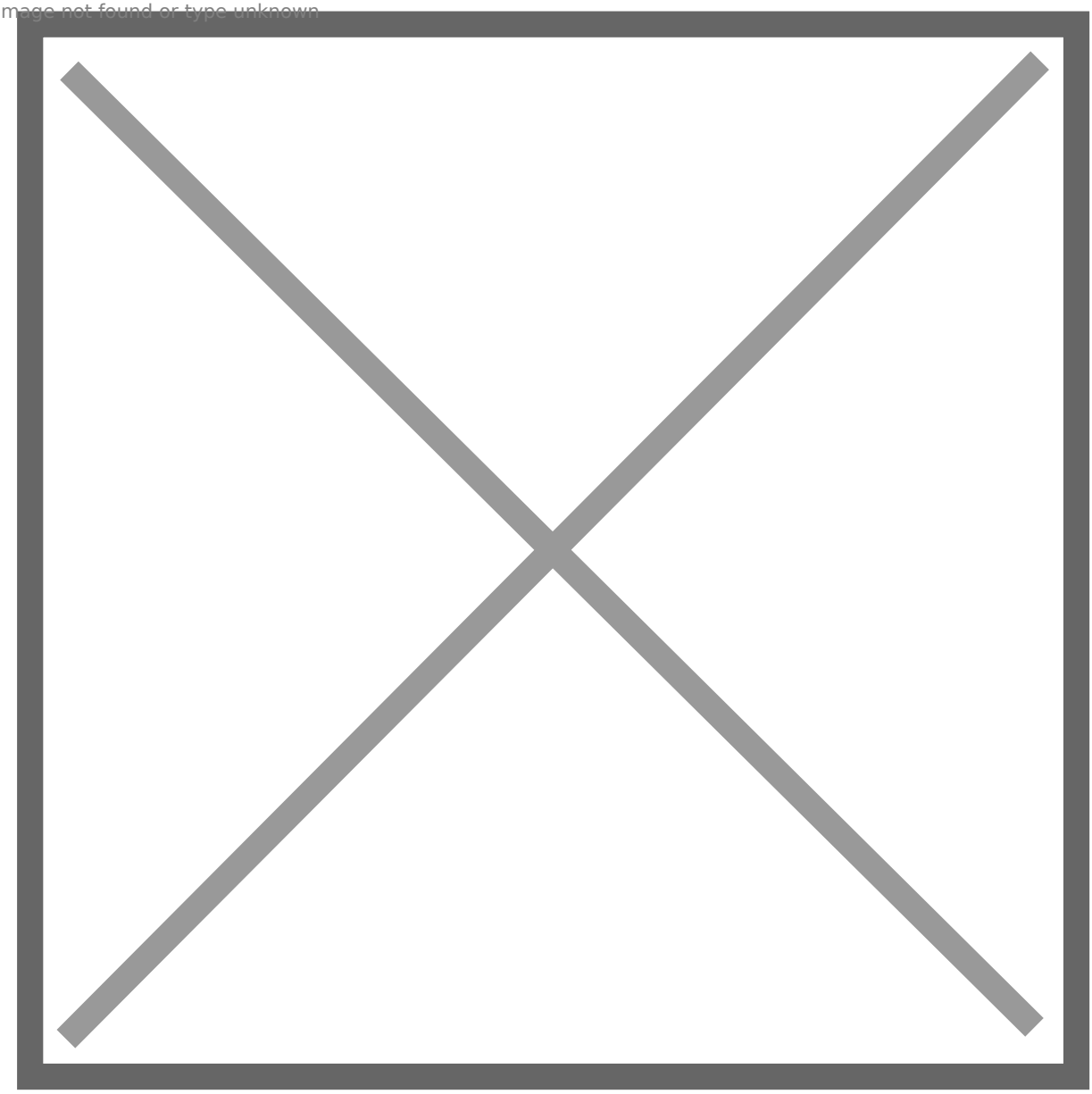
After that, we can choose to set SPN for svc_sql, which is designed as service account for SQL Server instances in domain. It is not required to set it now, but it does not hurt. Why? Because we can use a tool to automate this process later without getting any error. If you follow my SPN settings, please make sure your SQL Server instance are named DB01 and DB02 respectively. Later I will show how to configure SQL Server instances.

Image not found or type unknown



Then, I add helen.park to Helpdesk group. You can also create more groups, or add more users to groups.

Image not found or type unknown



helen.park should be able to **RDP** to **client01**, we need to add helen.park to a localgroup in client01, I will mention it later. jason.hudson has **RDP** and **WinRM** right to **SRV01**, so add him to two localgroups in SRV01. Therefore, we need to impersonate **jason.hudson** instead of Administrator when abusing delegation :). Instead of adding these users to local group, we can also link a GPO to them to enforce. This video shows detailed steps to achieve this:

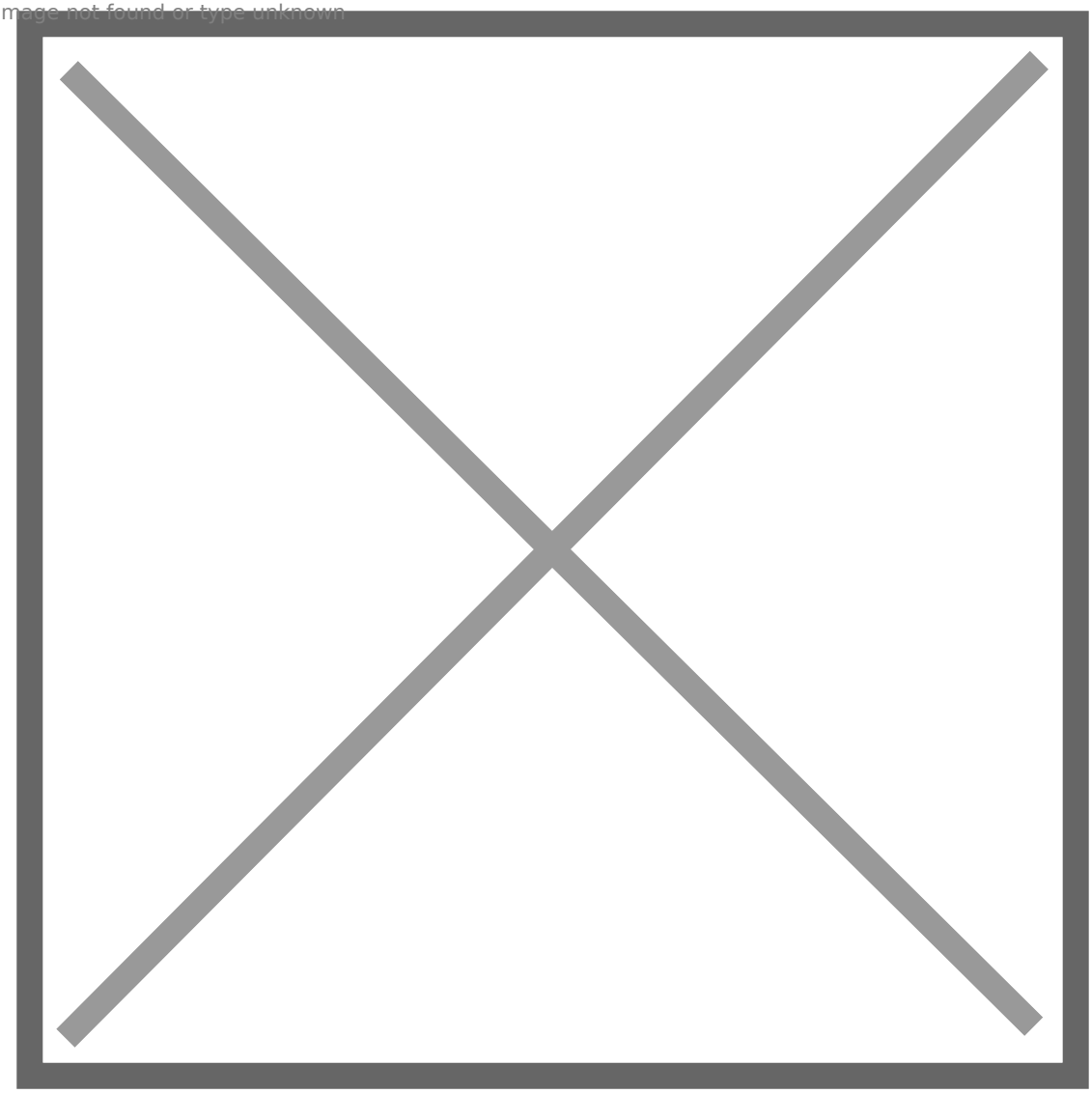
https://www.youtube.com/watch?v=euFiRyjRt1E

And I also turn on **automatic logon** for domain administrator on DC, you can check this article:

https://docs.microsoft.com/en-us/troubleshoot/windows-server/user-profiles-and-logon/turn-on-automatic-logon.

Besides, it is up to you whether turn on/off windows defender firewall. It is on by default, but I turn off it. **The setting is the same for every windows domain computers.**

Image not found or type unknown



Now, we completed basic settings on DC, but we will revisit DC after adding domain computers and configuration of SQL Server instances.

LINUX DOMAIN COMPUTER 1

web01.blackops.local

Since it is difficult to configure SRV01 and SRV02, so let’s start from easier ones.

First of all, we need to set DC’s ip as DNS. And add a new entity to /etc/resolve.conf. Be aware that after each reboot, we need to re-add the entity.

Image not found or type unknown

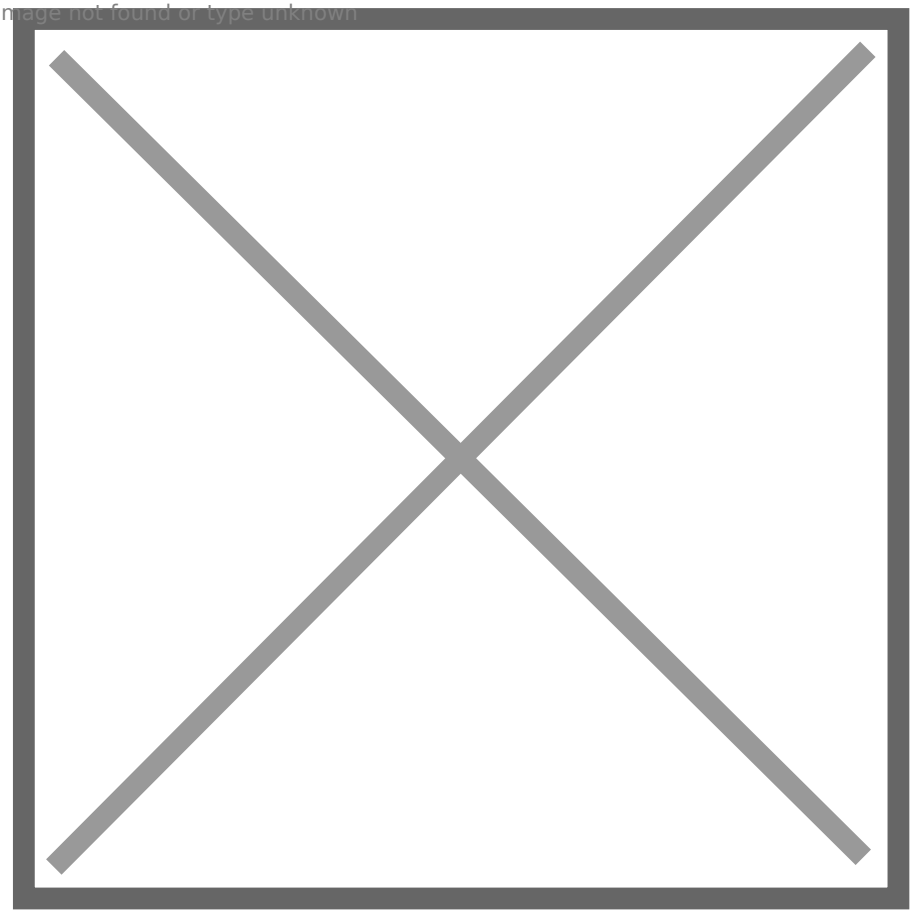
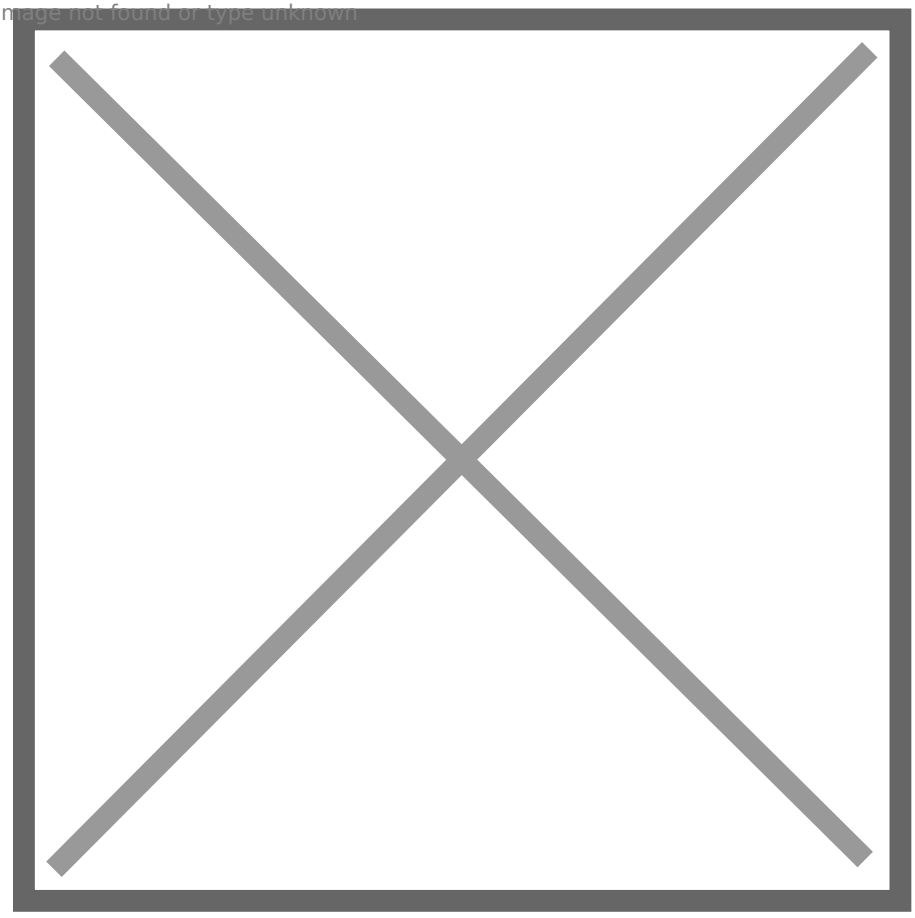
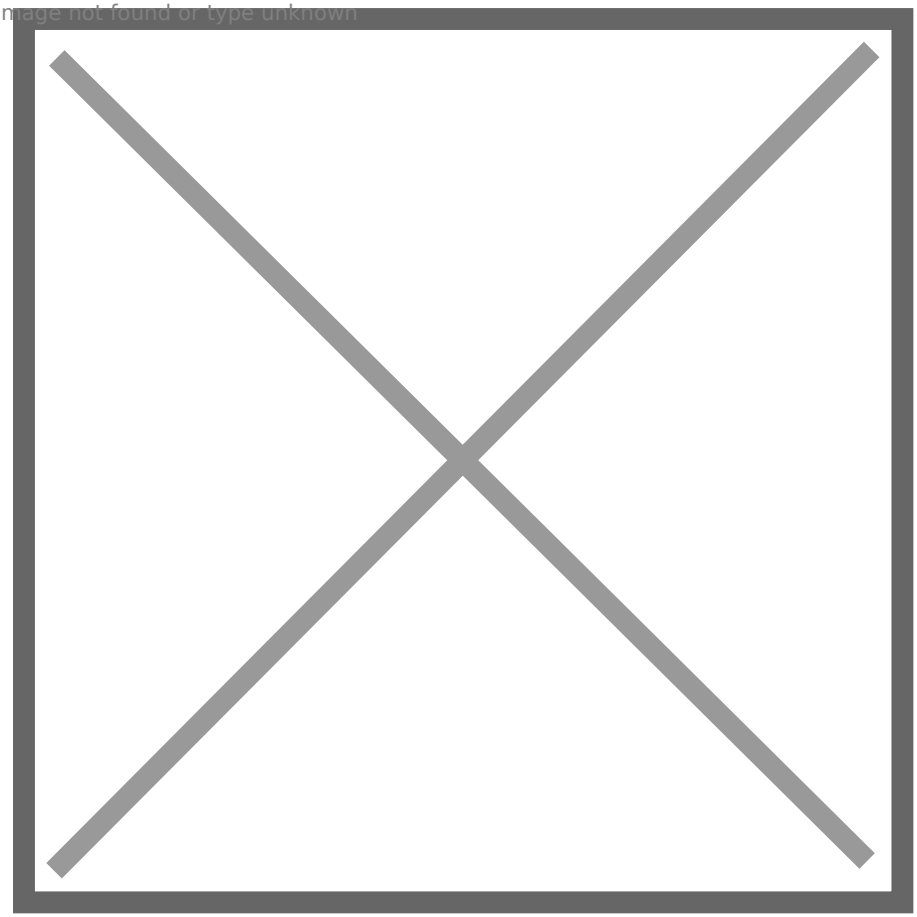


Image not found or type unknown



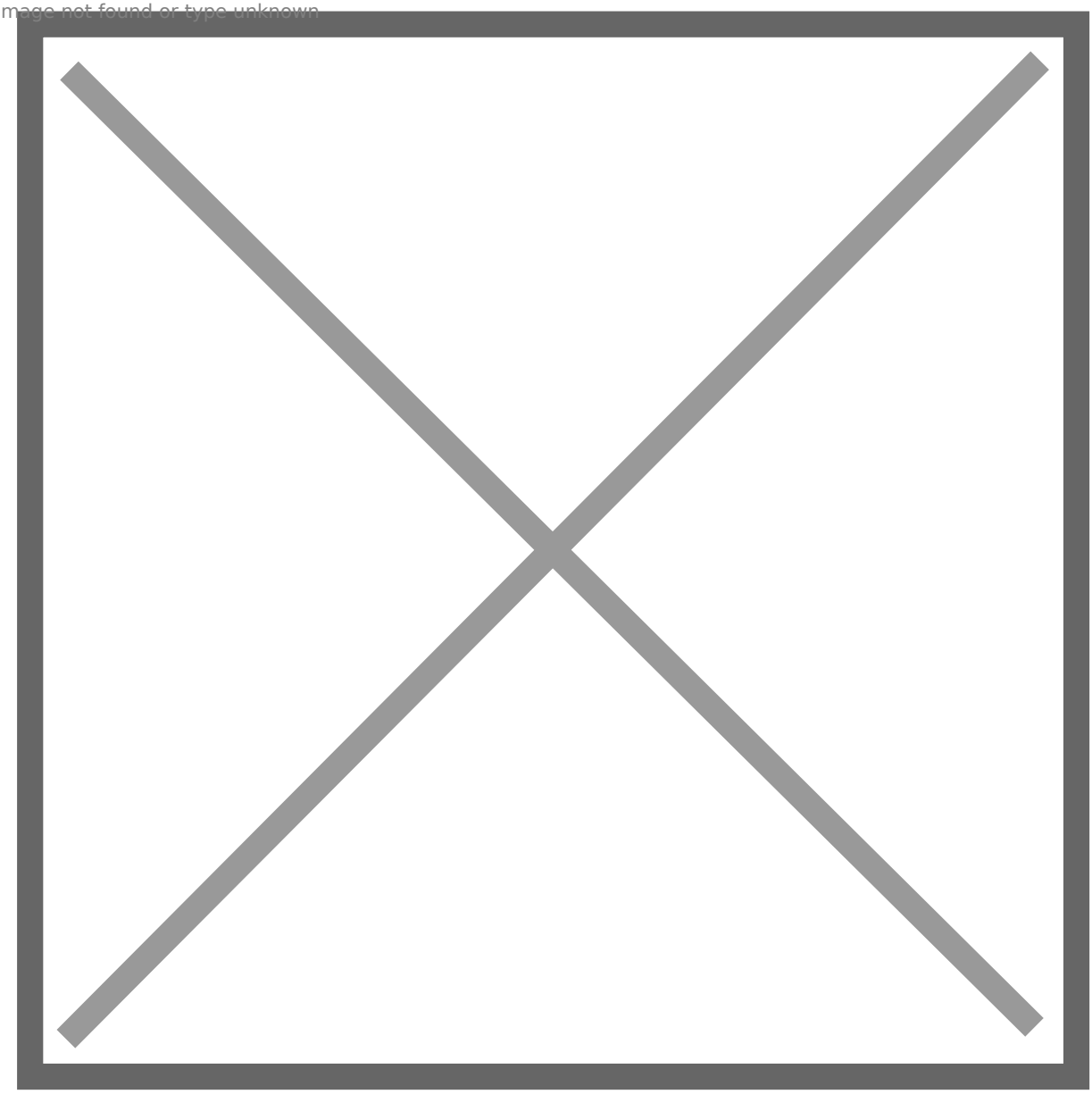
By this way, we can look up domain computers and join domain later.

Image not found or type unknown



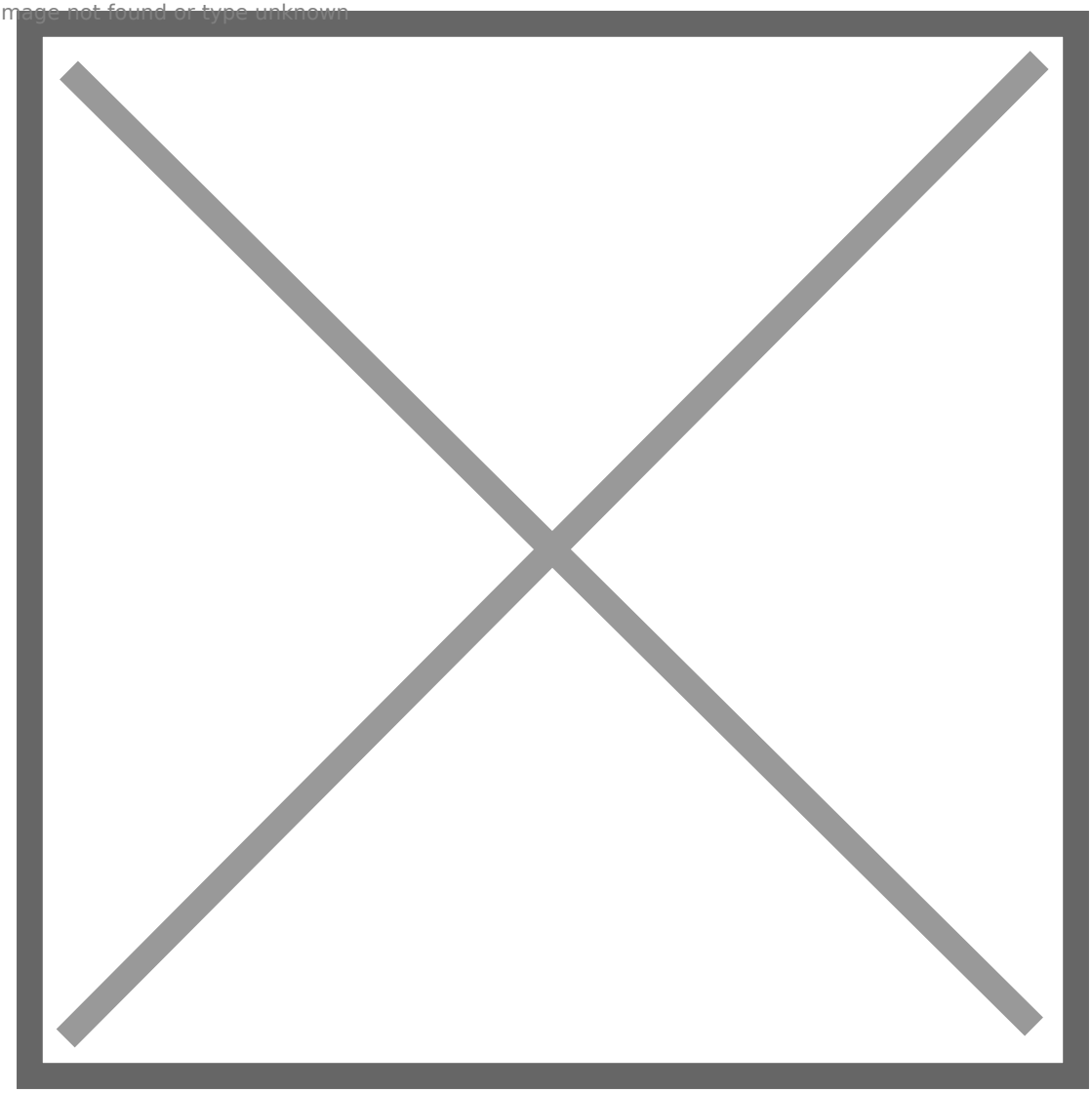
Then we need to set up few Linux local accounts.

Image not found or type unknown



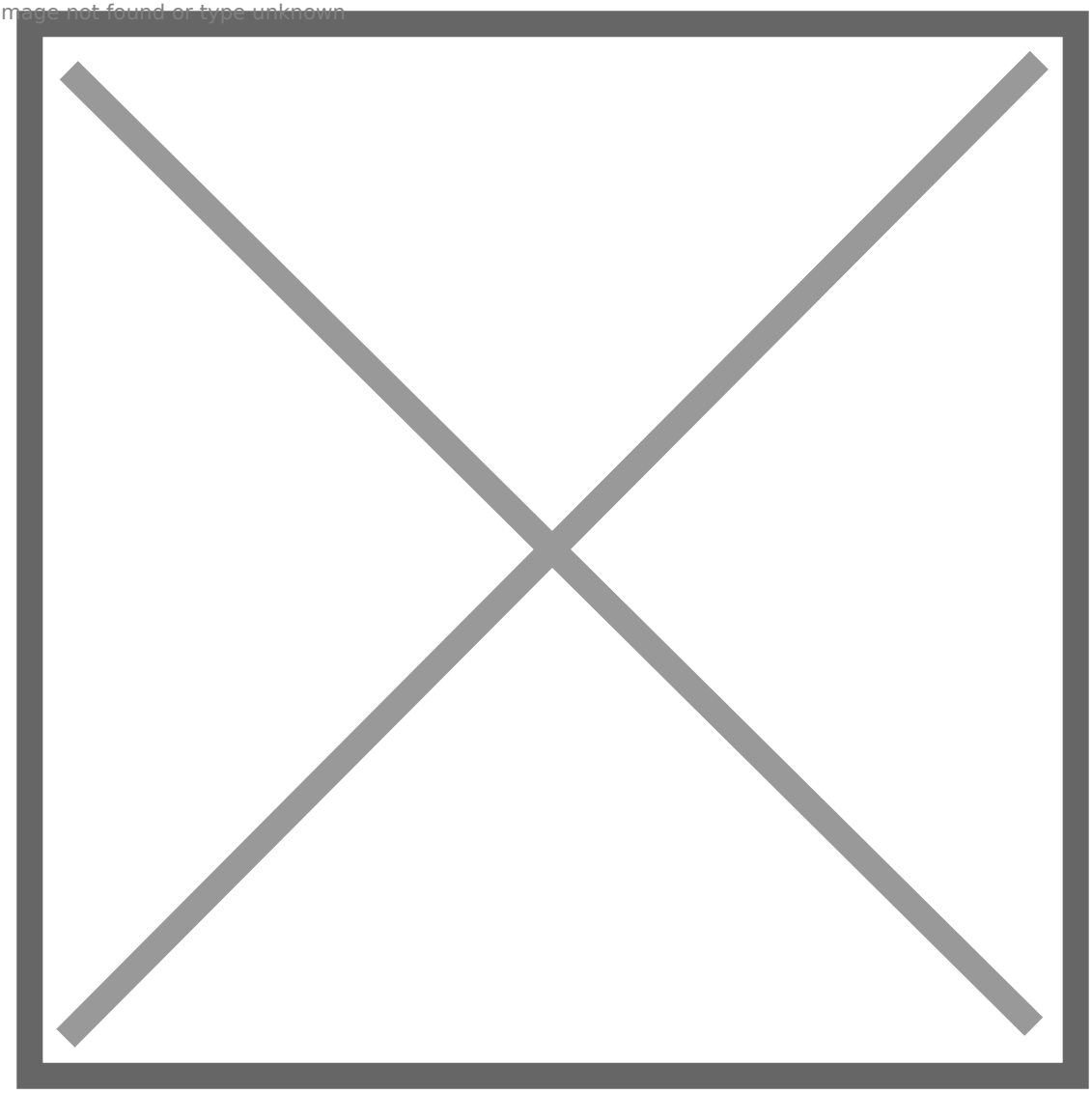
In regard to how to join a Linux computer to domain, this article gives detailed instruction: https://www.informaticar.net/join-ubuntu-machine-to-windows-domain/. You will not make any mistake as long as you follow steps. You can use **klist** to check tickets to verify that the Linux machine successfully joined domain.

Image not found or type unknown



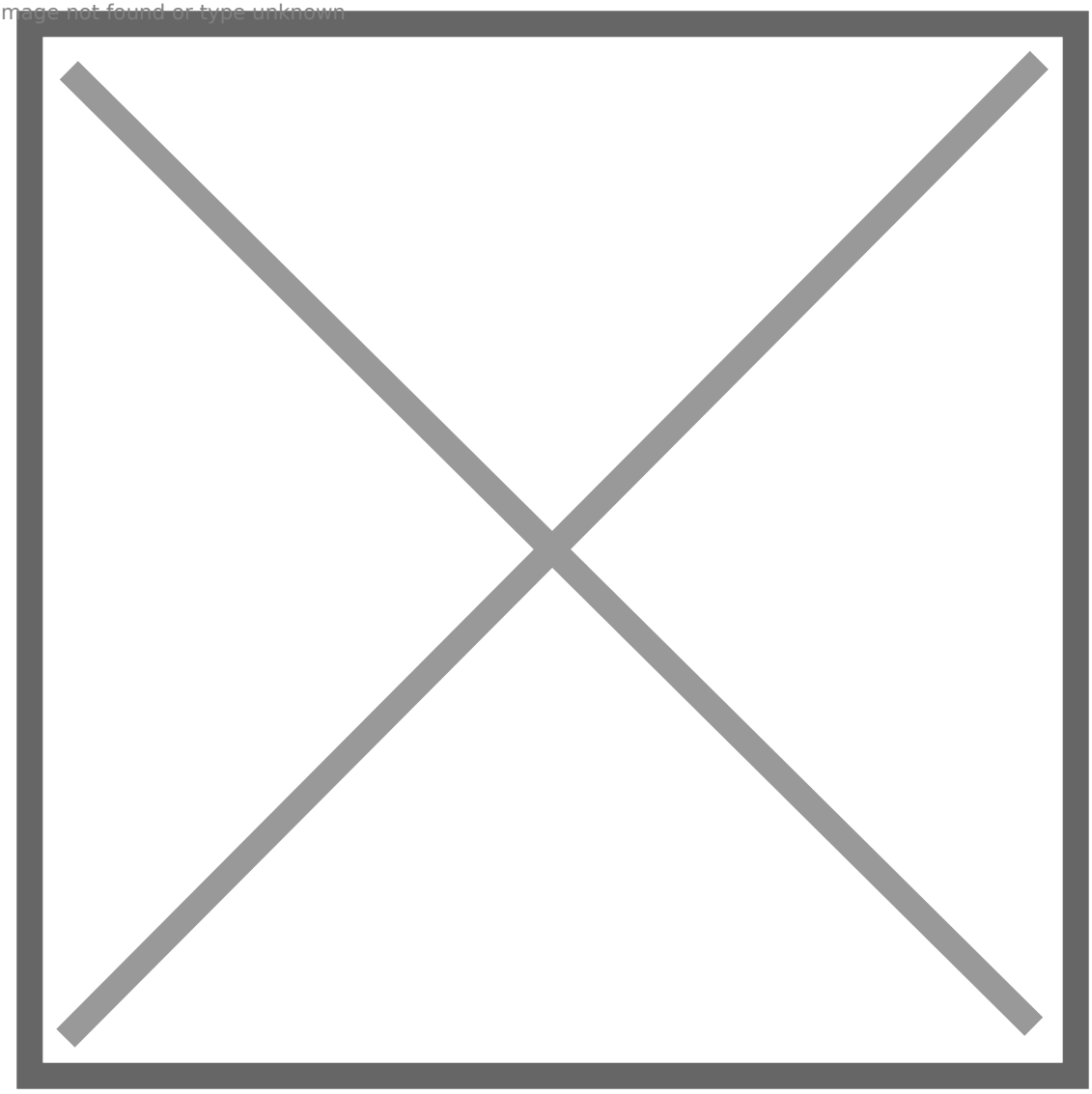
File **/etc/krb5.keytab** is readable for root by default, it contains machine account web01$’s credential. We can use python script keytabextract.py (https://github.com/sosdave/KeyTabExtract) to extract them.

Image not found or type unknown



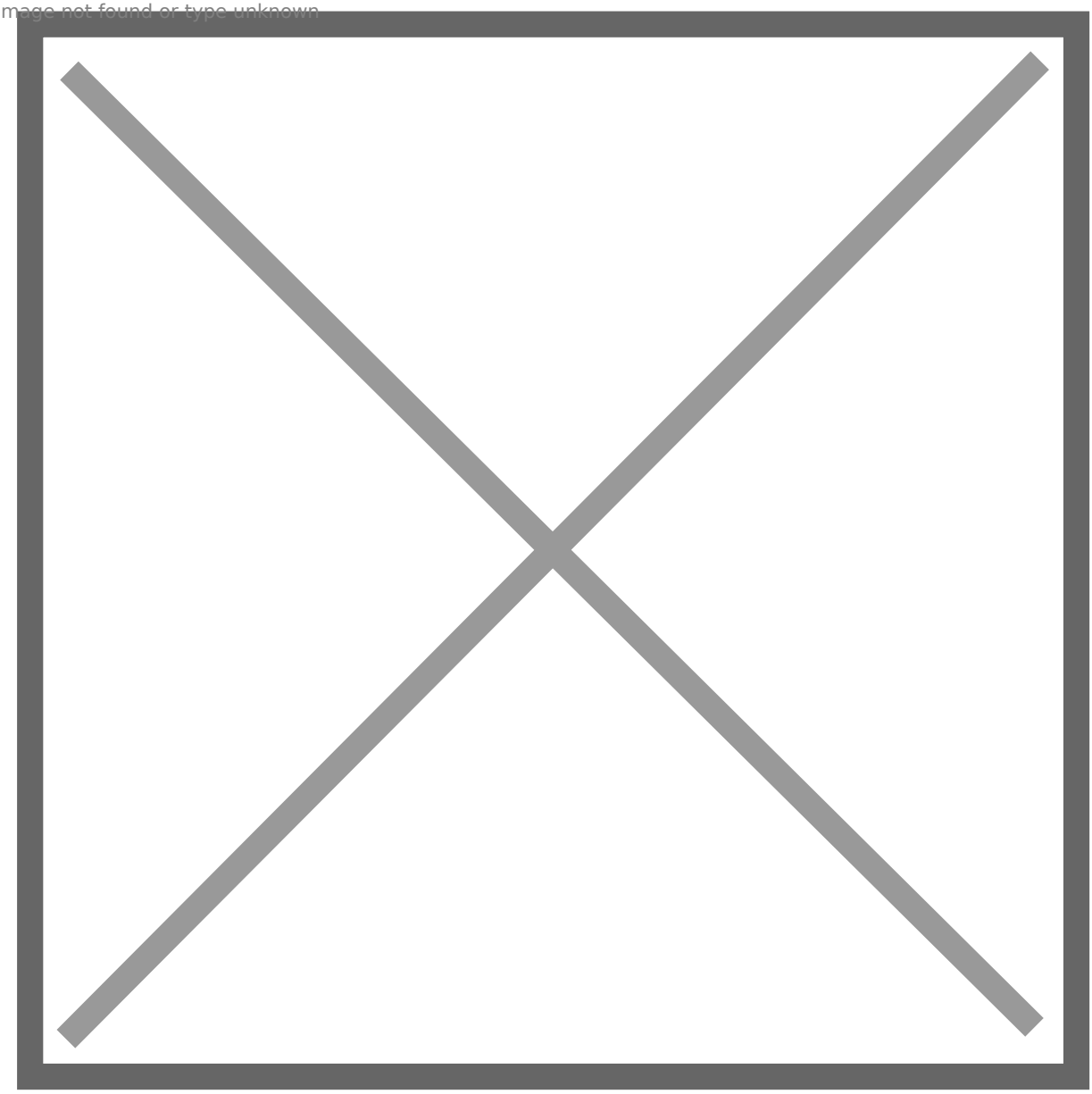
After gaining root, or even if we can read it as a normal user, we can use credential **web01$:5db7a1891649cef400f8cd6923bb4a69** to authenticate to domain to have a domain context or enumerate domain information. One example is to use bloodhound-python to collect domain information.

Image not found or type unknown



Okay, we have successfully added web01 to domain, we can use the exact same steps to add file01 to domain. Now, we need to deploy vulnerable services/app, and rabbit holes lol. The following table reflect my design.

Image not found or type unknown



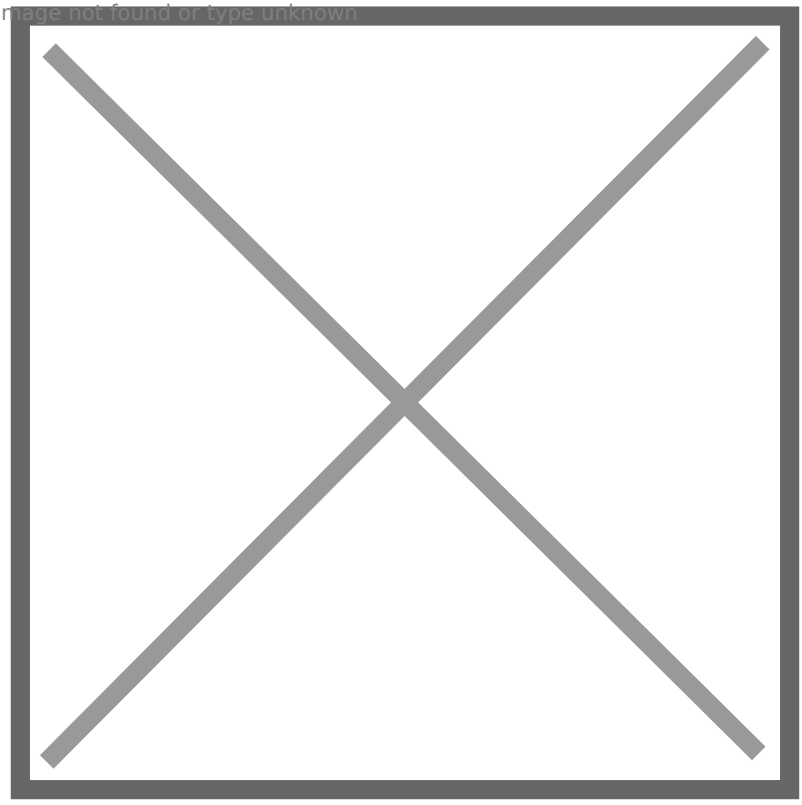
There are multiple apps/services to be installed and configured, you can check following links to follow steps.

Port 22: SSH

Add a line to **/etc/ssh/sshd_configure**:

Denyusers mailadmin

Image not found or type unknown

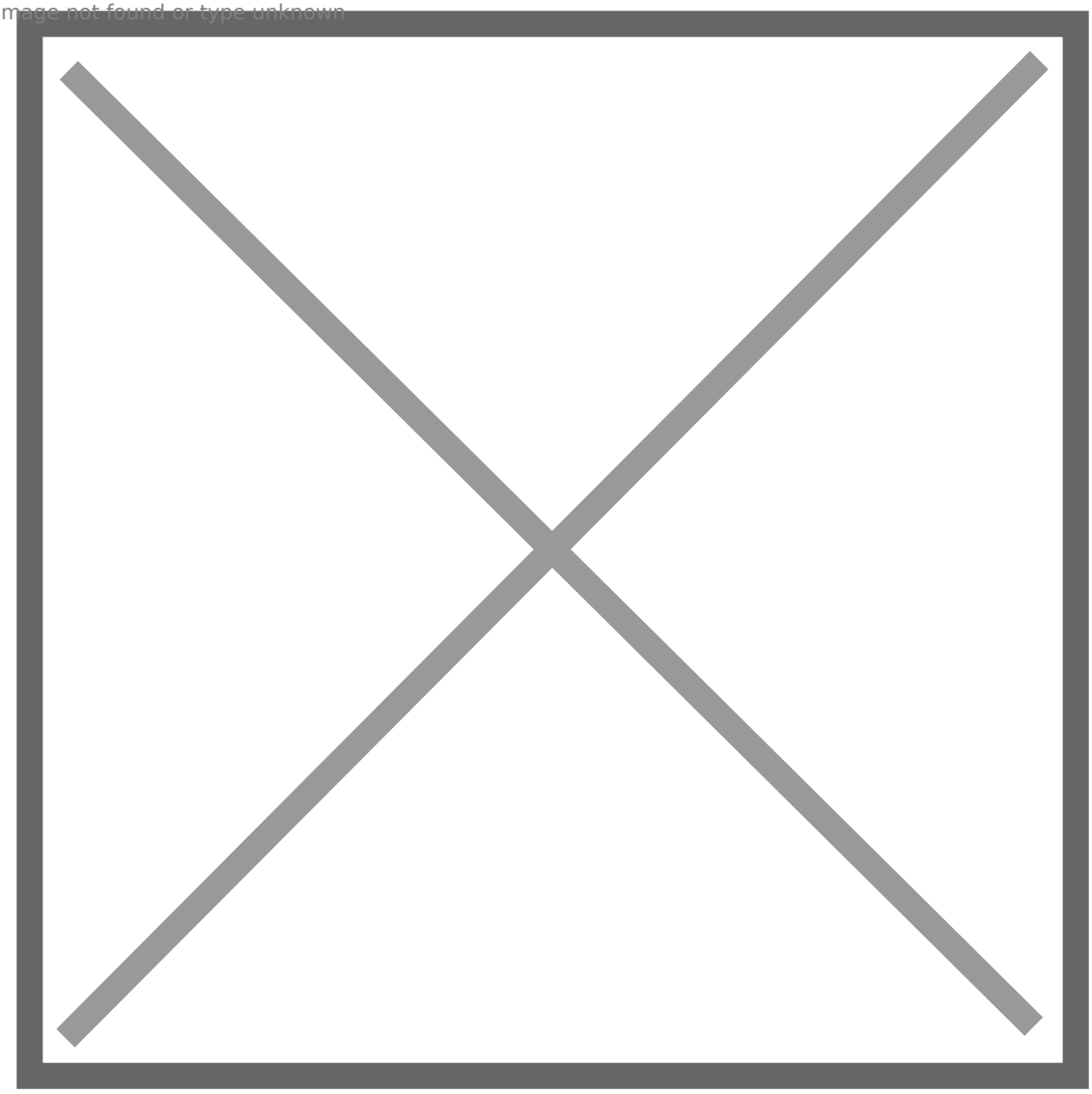


This step is to deny mailadmin’s SSH access, since mailadmin has a weak password. It should be like a service account.

Port 25: Postfix SMTP: https://ubuntu.com/server/docs/mail-postfix

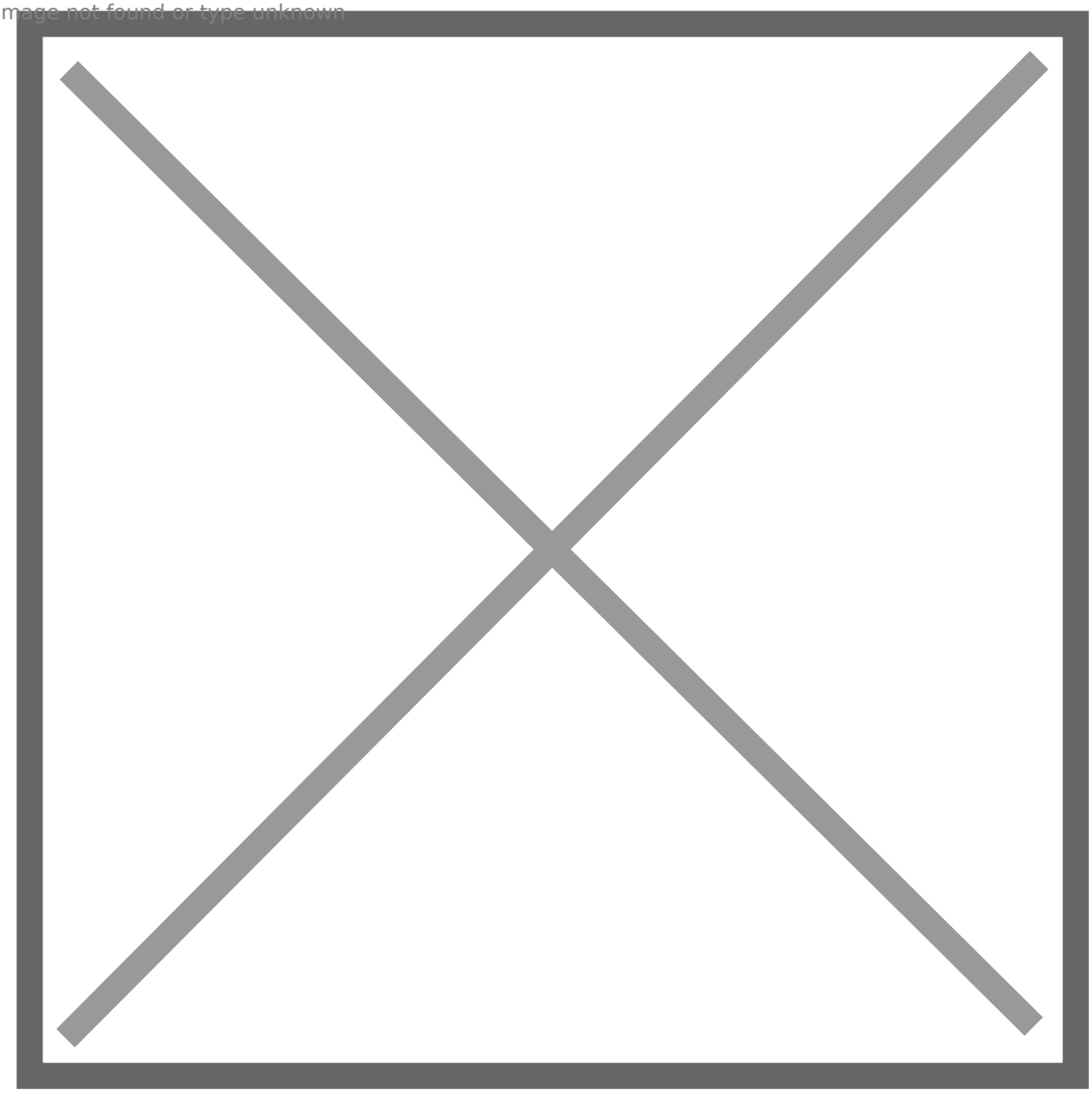
To make it simple, we can **stop at** SMTP Authentication section.

Image not found or type unknown



And then, we need to send an email via SMTP, check commands in the screenshot.

Image not found or type unknown

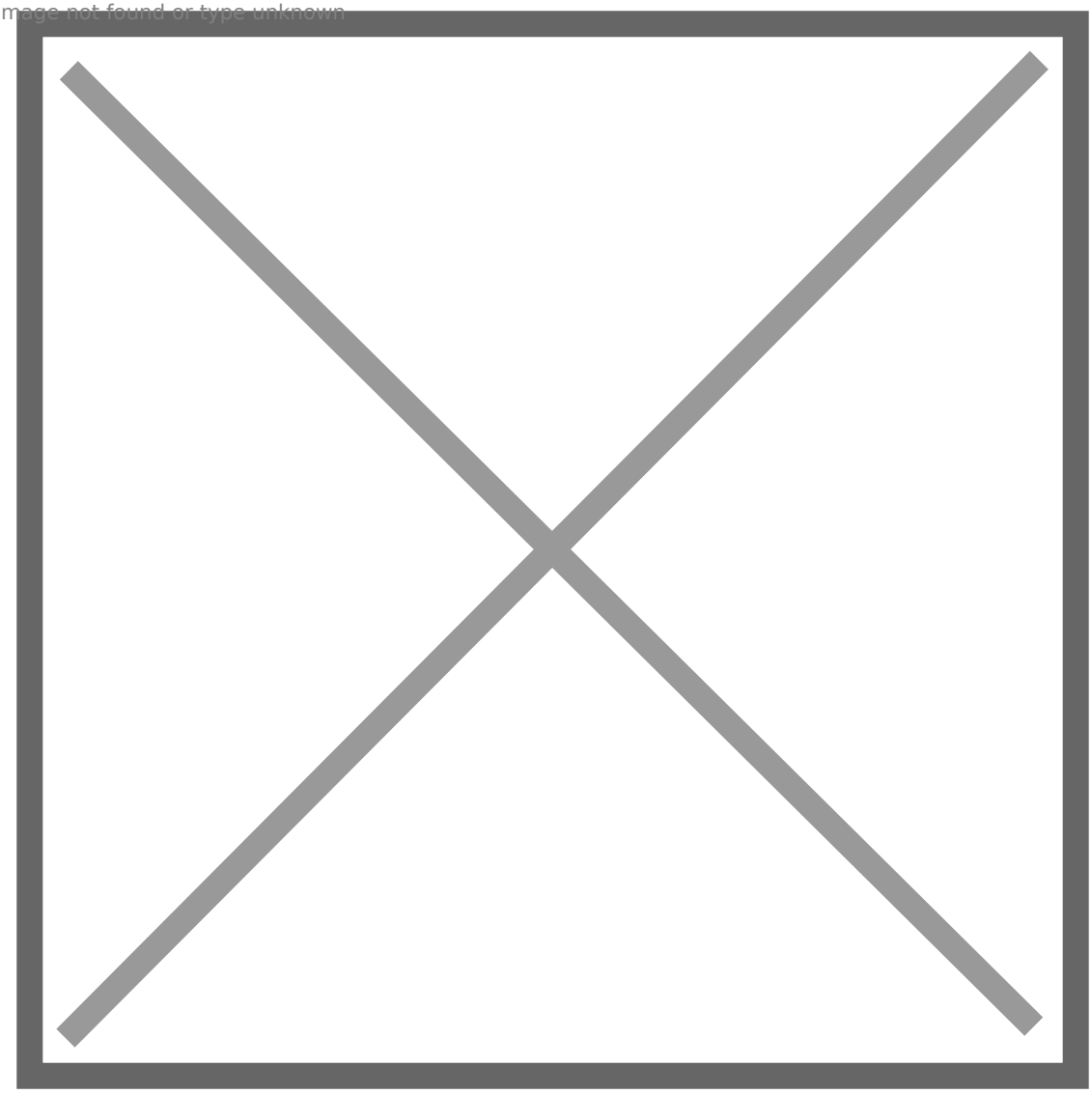


So the email will be delivered to mailadmin’s inbox.

Port 110 and 143: Dovecot (POP3+IMAP): https://ubuntu.com/server/docs/mail-dovecot

To make it simple, we can **stop at** Dovecot SSL Configuration section.

Image not found or type unknown



And we need to allow plaintext authentication to POP3 server, just append two lines to **/etc/dovecot/dovecot.conf**:

disable_plaintext_auth=no

ssl=yes

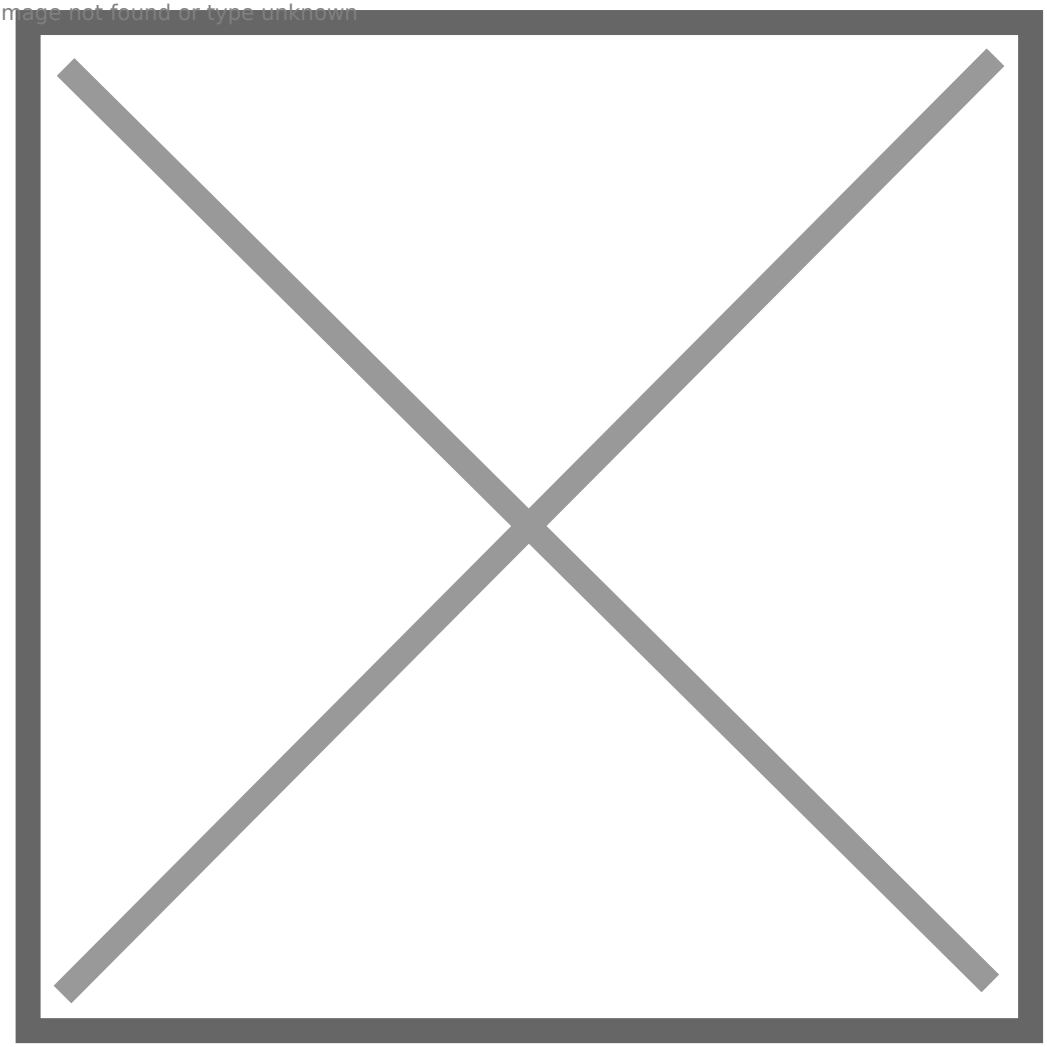
Then we can log in POP3 server, otherwise we cannot authenticate to POP3 server.

Port 80: WordPress: https://ubuntu.com/tutorials/install-and-configure-wordpress#1-overview

It is simple, just follow steps in this link. After completing the installation, register 2 users: mason, hudson.

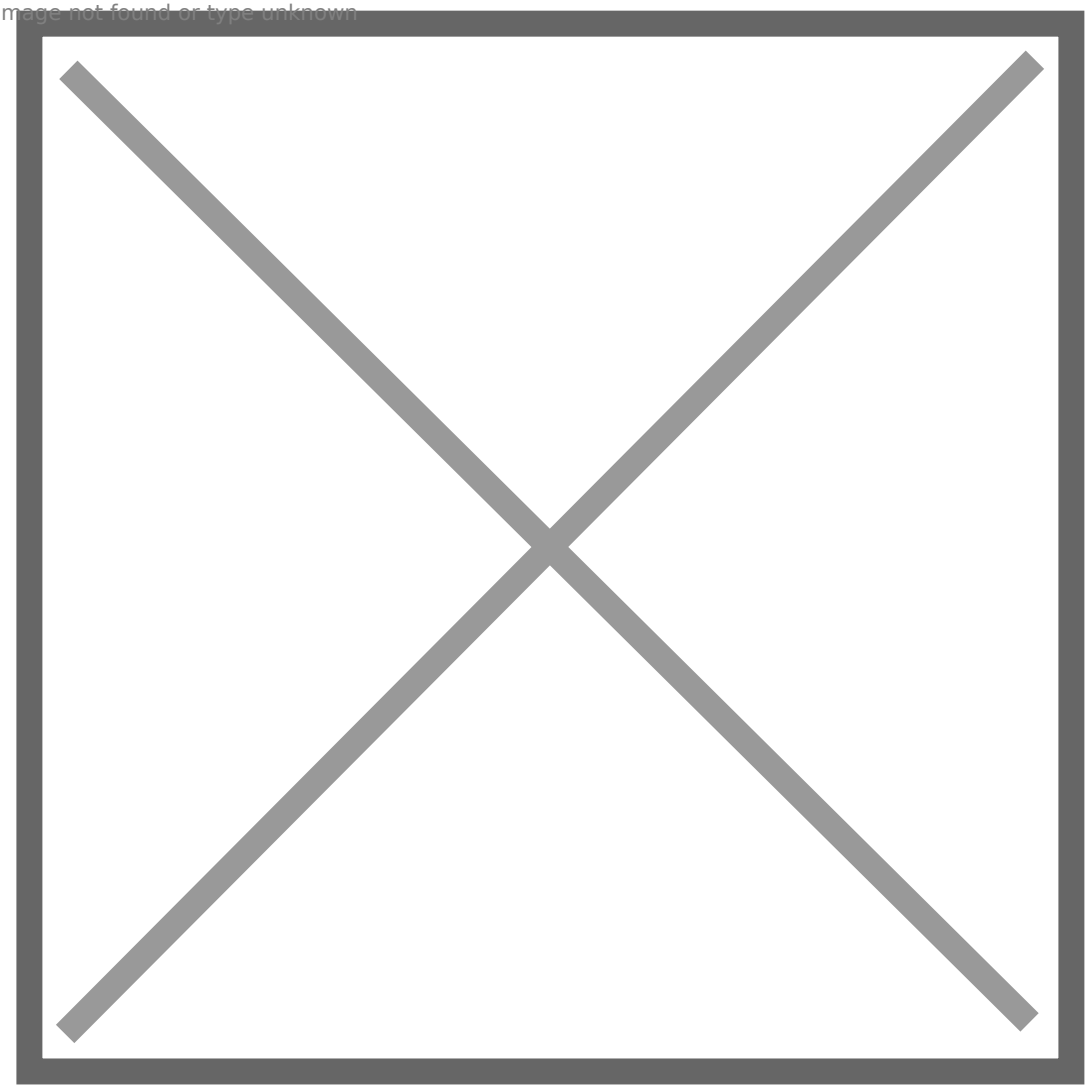
Log in as mason, and post an article like this:

Image not found or type unknown



Then log in as hudson, leave a comment. This is an indicator that mason manages mailadmin account, and this account has weak password: Password. After that, log in as mason or admin to approve hudson’s comment. Otherwise, hudson’s comment will not be displayed.

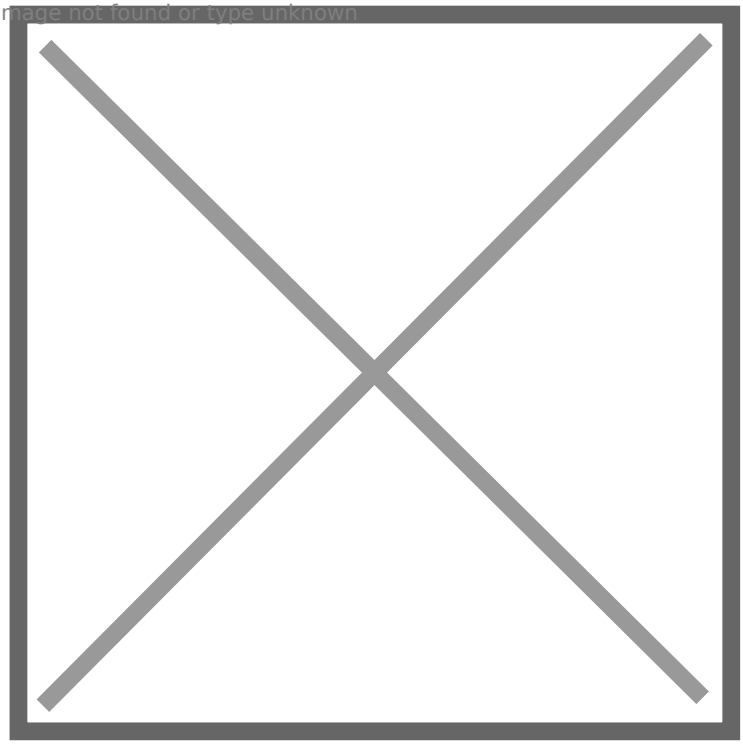
Image not found or type unknown



Port 445: Samba

Create a new folder /var/backups/www/html, and copy /var/www/html/wordpress to the new folder, and create a new share to map to this folder.

Image not found or type unknown



Do not forget to assign proper ownership and permission, otherwise the attacker cannot upload or read a file, so he will not upload a shell and fall into the rabbit hole lol

Image not found or type unknown

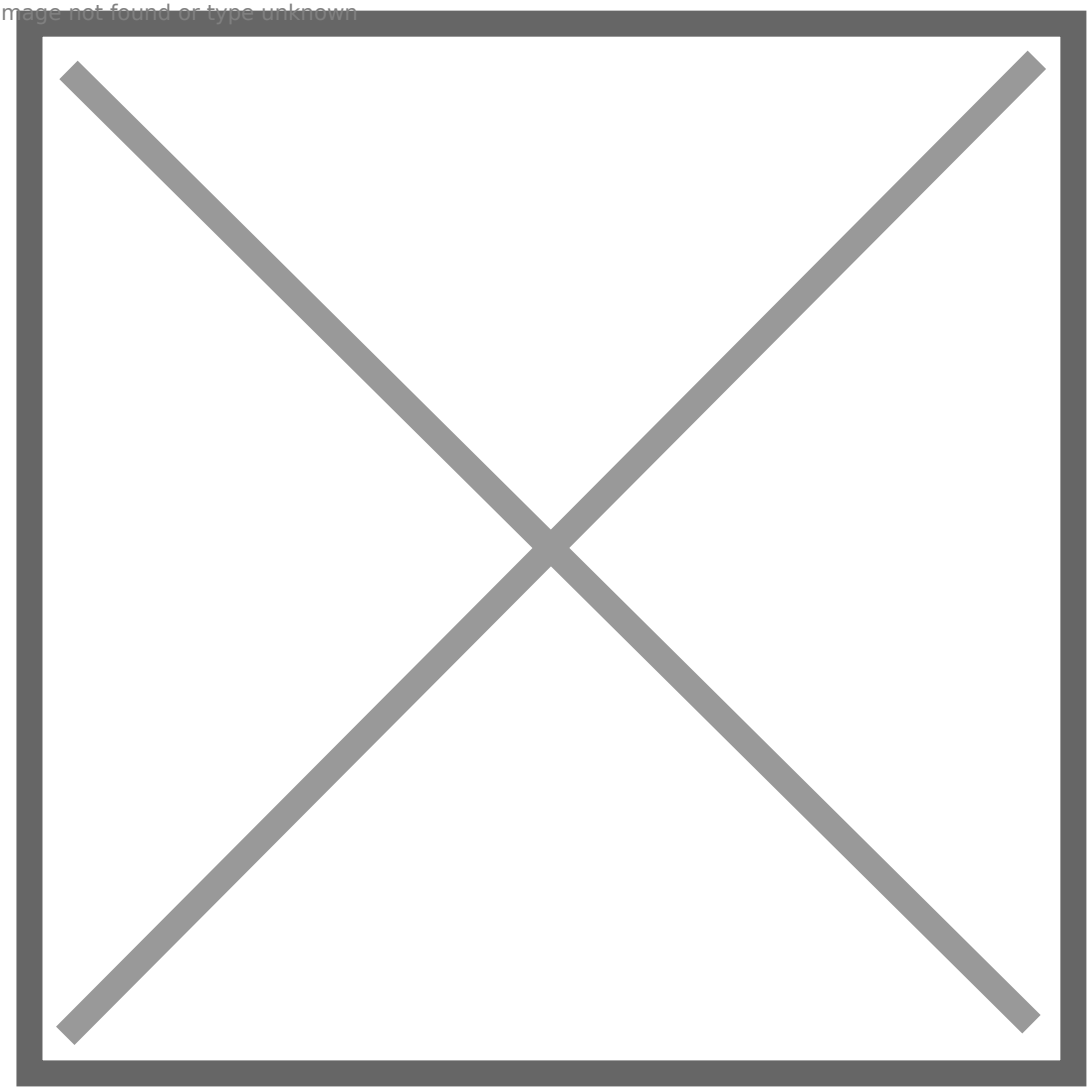
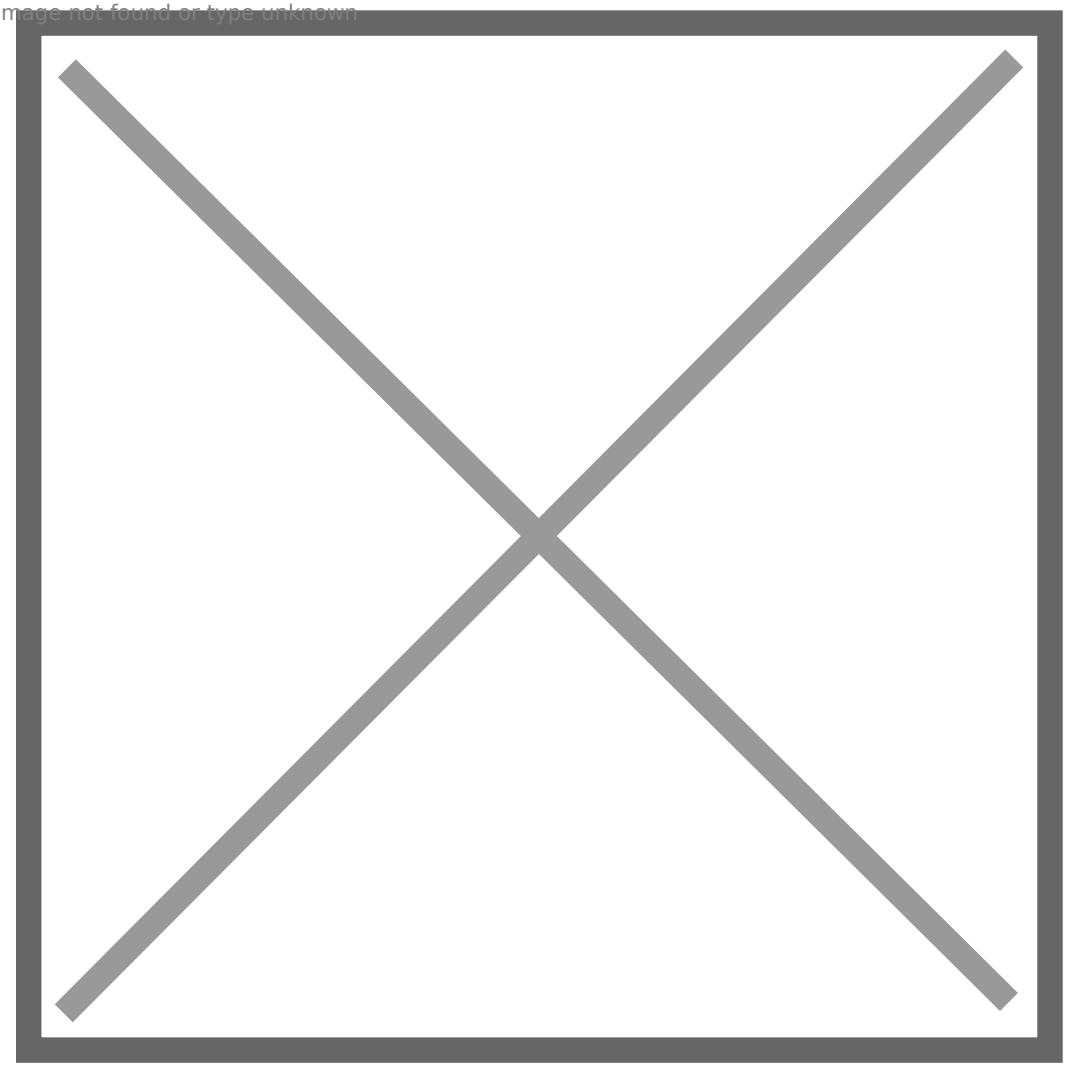


Image not found or type unknown



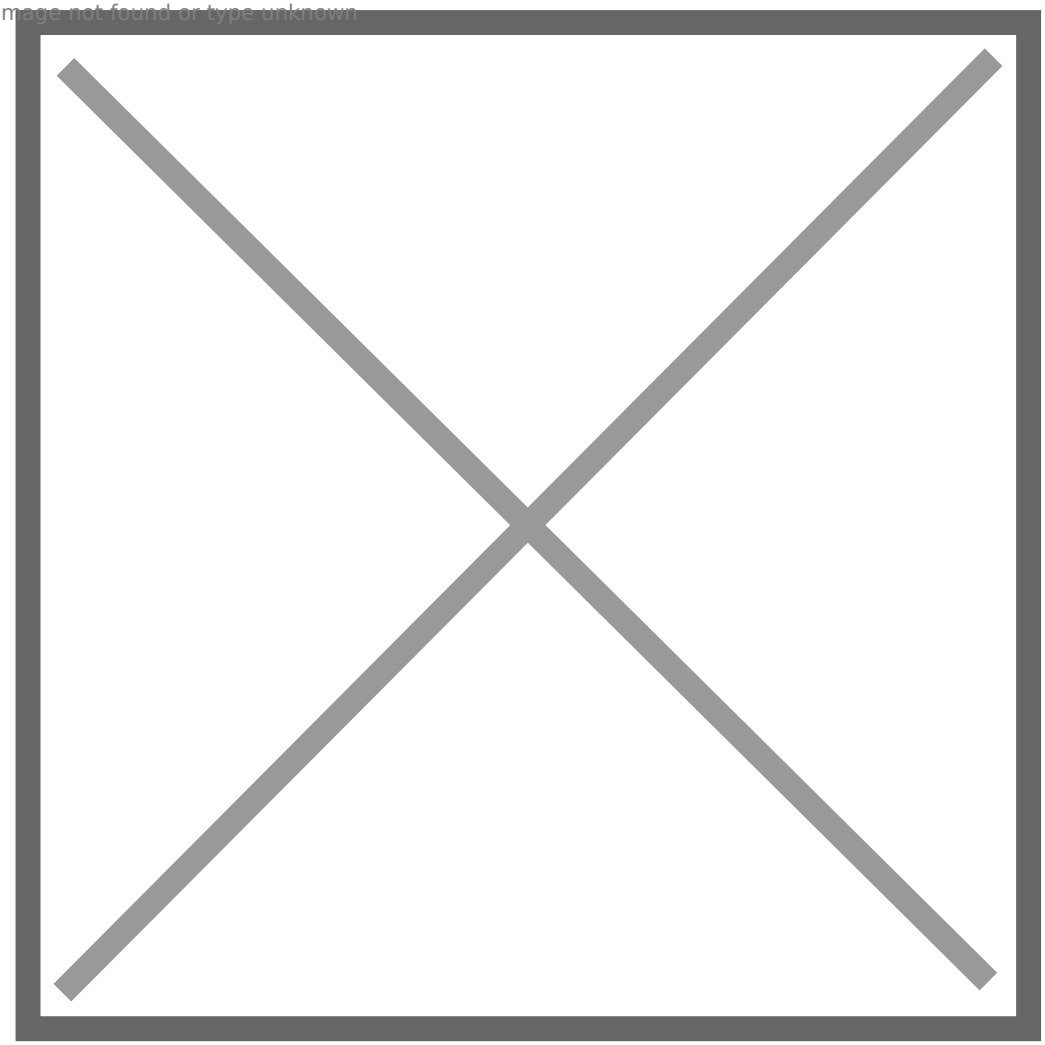
Port 5601: Kibana 6.5

Download link: https://www.elastic.co/cn/downloads/past-releases/kibana-6-5-0

Download **deb 64-bit**, and then use dpkg to install it, it is very simple.

But do not forget to edit /etc/kibana/kibana.yml to uncomment few lines and change server.hot to 0.0.0.0.

Image not found or type unknown



This version of kibana is vulnerable to a RCE vulnerability, you can find the PoC here: https://github.com/mpgn/CVE-2019-7609

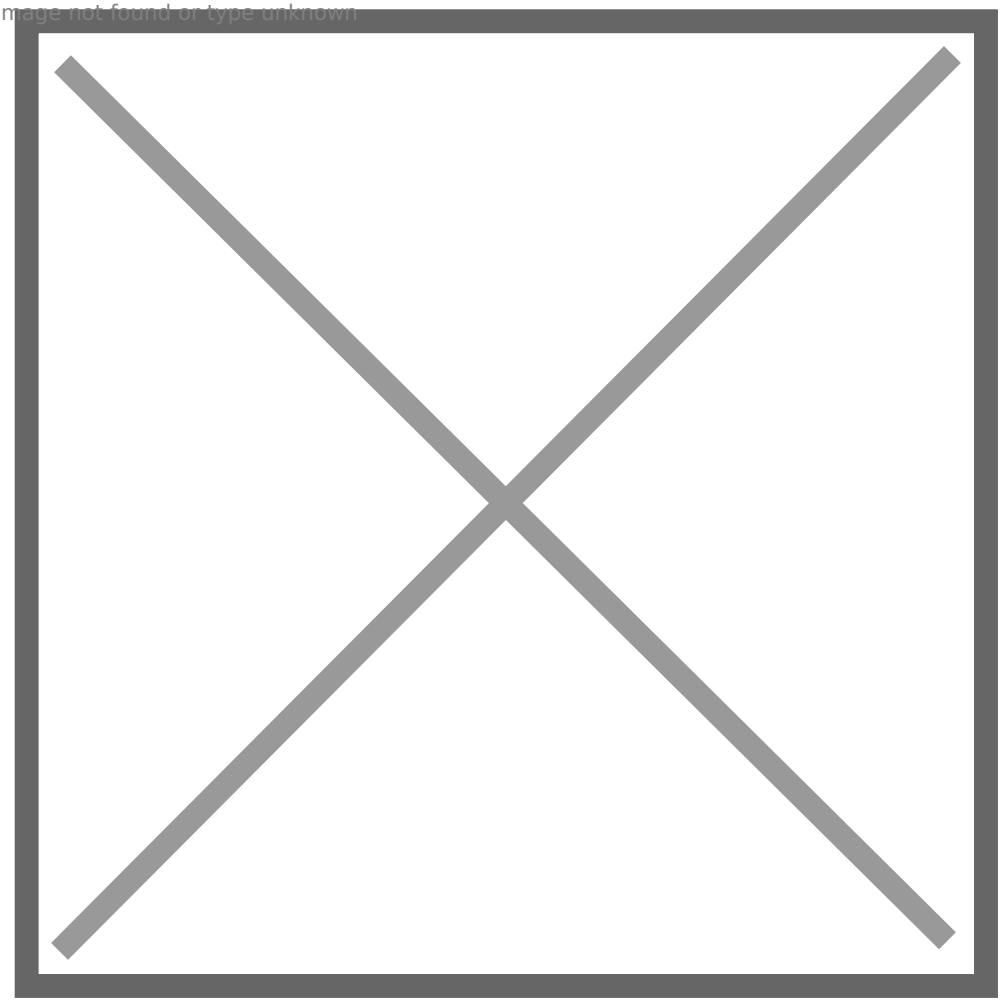
Follow the steps, and you can get a shell as kibana. But unfortunately, there is no intended privilege escalation vector for user kibana, though I am not sure if all Nday vulnerabilities have been fixed. Therefore, it is a rabbit hole.

Port 9200: Elasticsearch 6.6

Download and install Elasticsearch 6.6 from https://www.elastic.co/cn/downloads/past-releases/elasticsearch-6-6-0 like how we installed Kibana, but we do not need to customize it.

Now, we almost finished. The last step is to grant mason a privilege to execute find with sudo permission without password. So only user mason can escalate our self to root and read /etc/krb5.keytab.

Image not found or type unknown



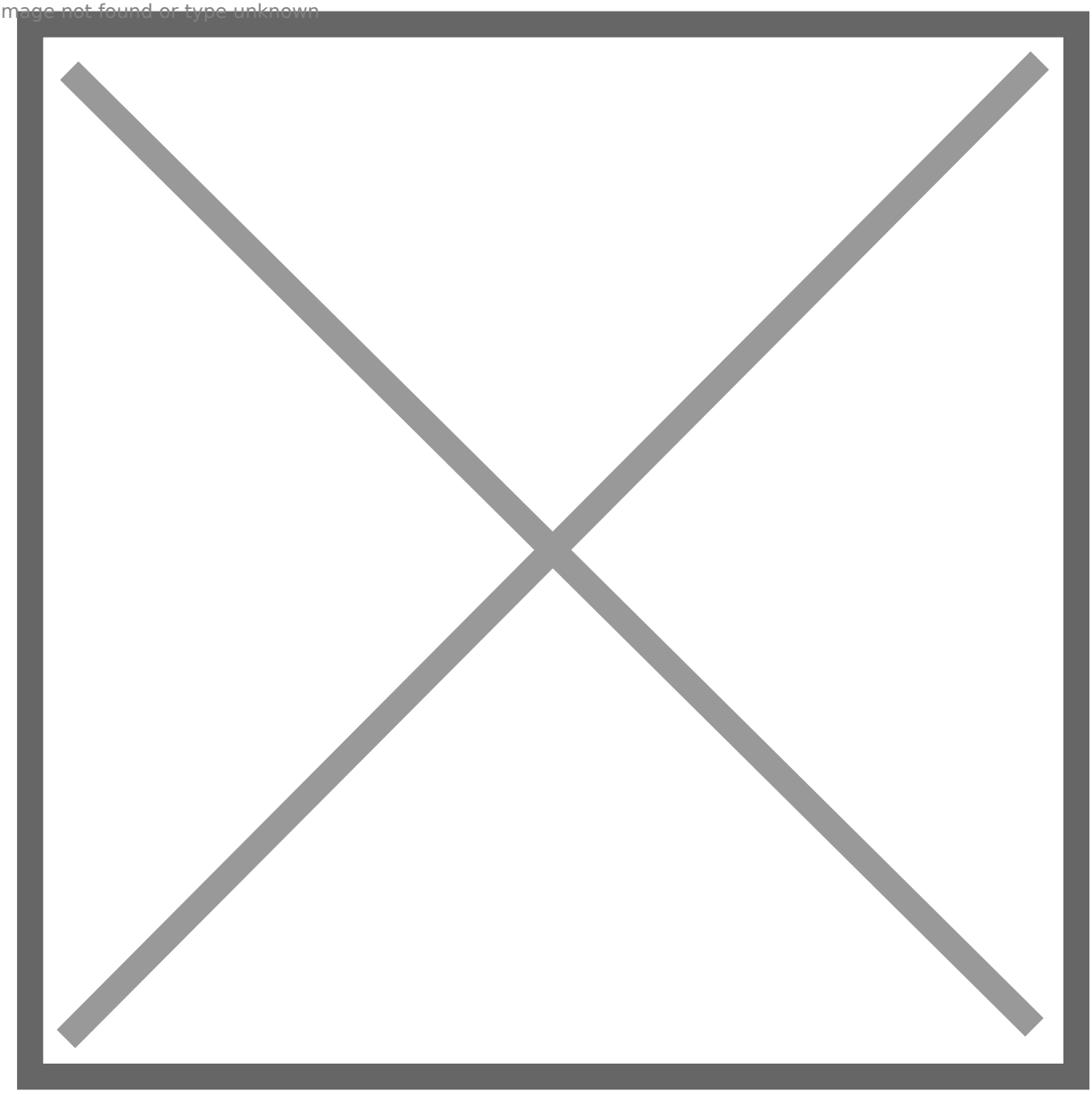
After knowing that alex.mason is a domain user, we should be aware that linux local user mason could share the same password with domain user alex.mason, so we can use SSH to move to file01 as alex.mason@blackops.local.

LINUX DOMAIN COMPUTER 2

file01.blackops.local

This linux machine is easier to configure. First, we need to set DC’s IP as DNS, and join file01 to domain, just as we previously did. We only need to configure FTP and add one user.

Image not found or type unknown



There is nothing too much to configure FTP. Use apt to install vsftpd. Then add helen as a linux local user. When we can use helen’s credential to authenticate to FTP server.

Image not found or type unknown

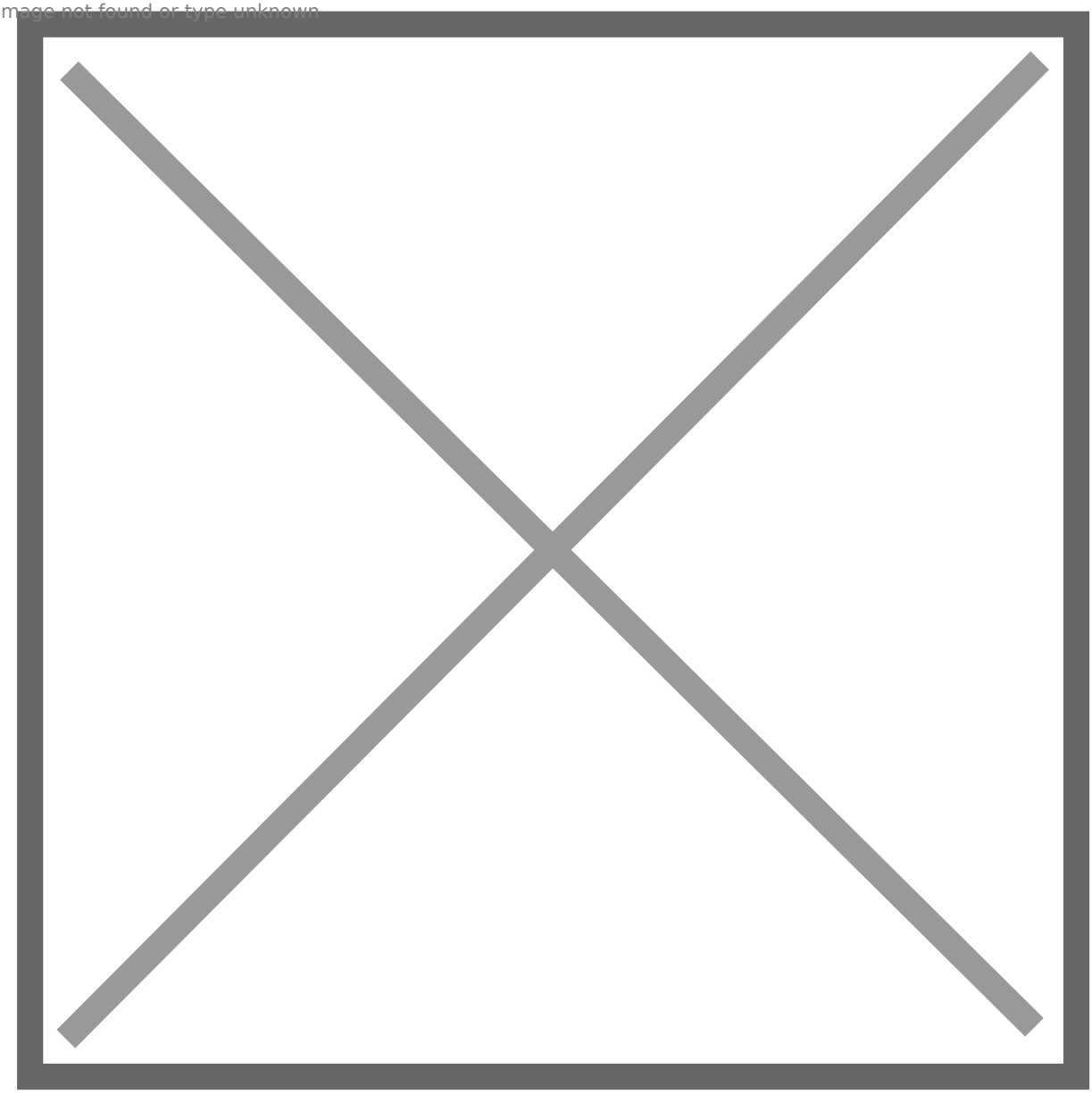
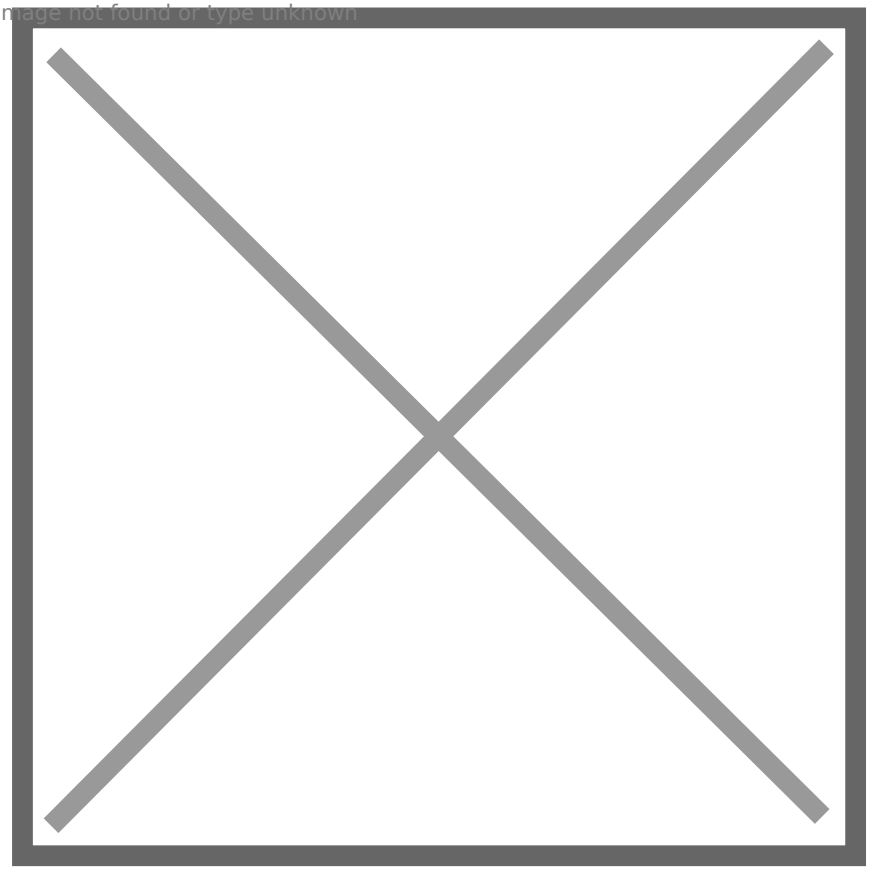


Image not found or type unknown



Many people only care about how to become root, and this is the reason why I make privilege escalation simple, I set multiple common binaries (cat, nc, find, etc.) SUID permission, and I also set tcpdump SUID. If check memo.txt, we can know that Helen keeps authenticating to FTP server. Since FTP does not have encryption, so we can use tcpdump to capture plaintext credential.

Image not found or type unknown

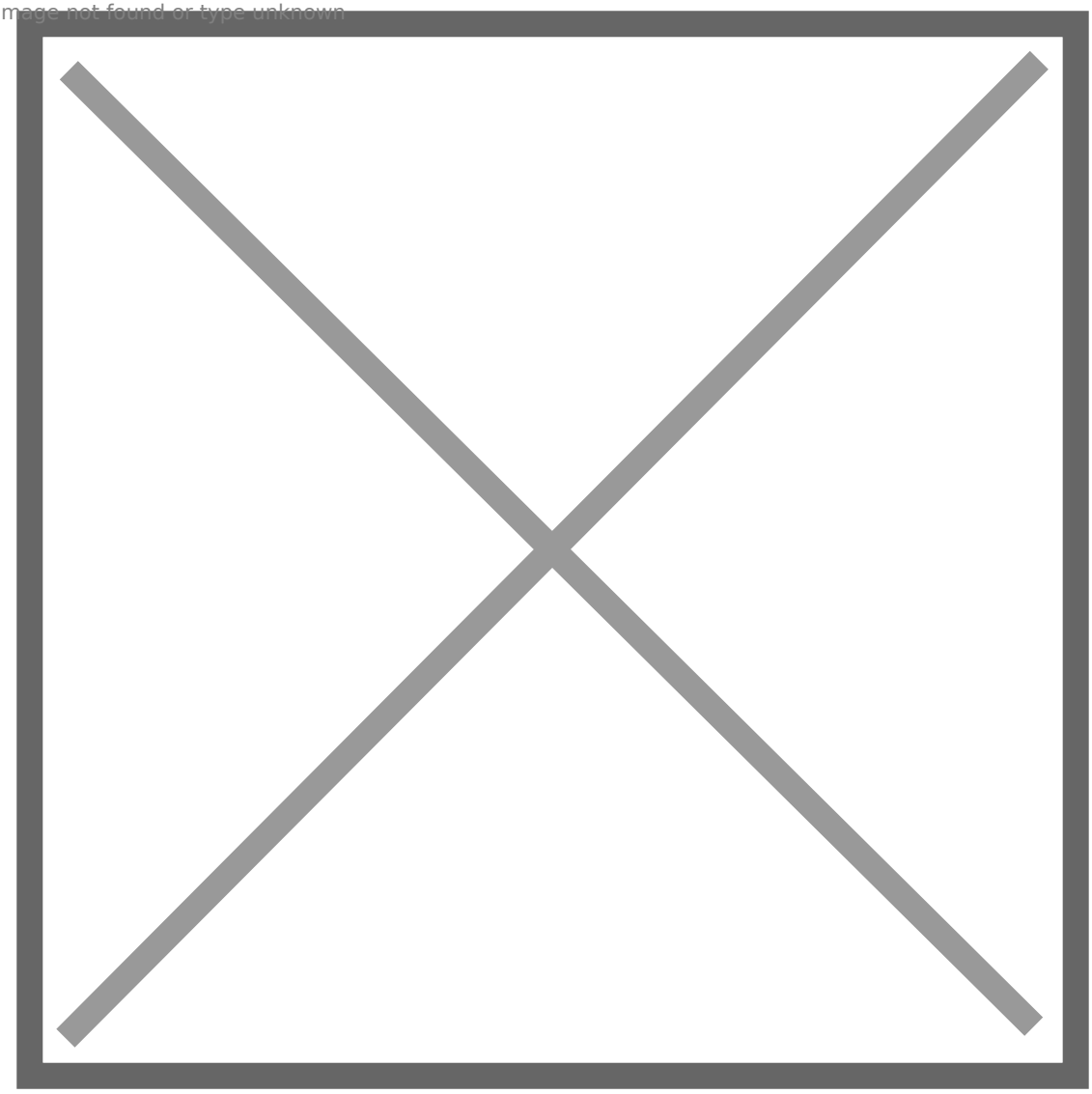
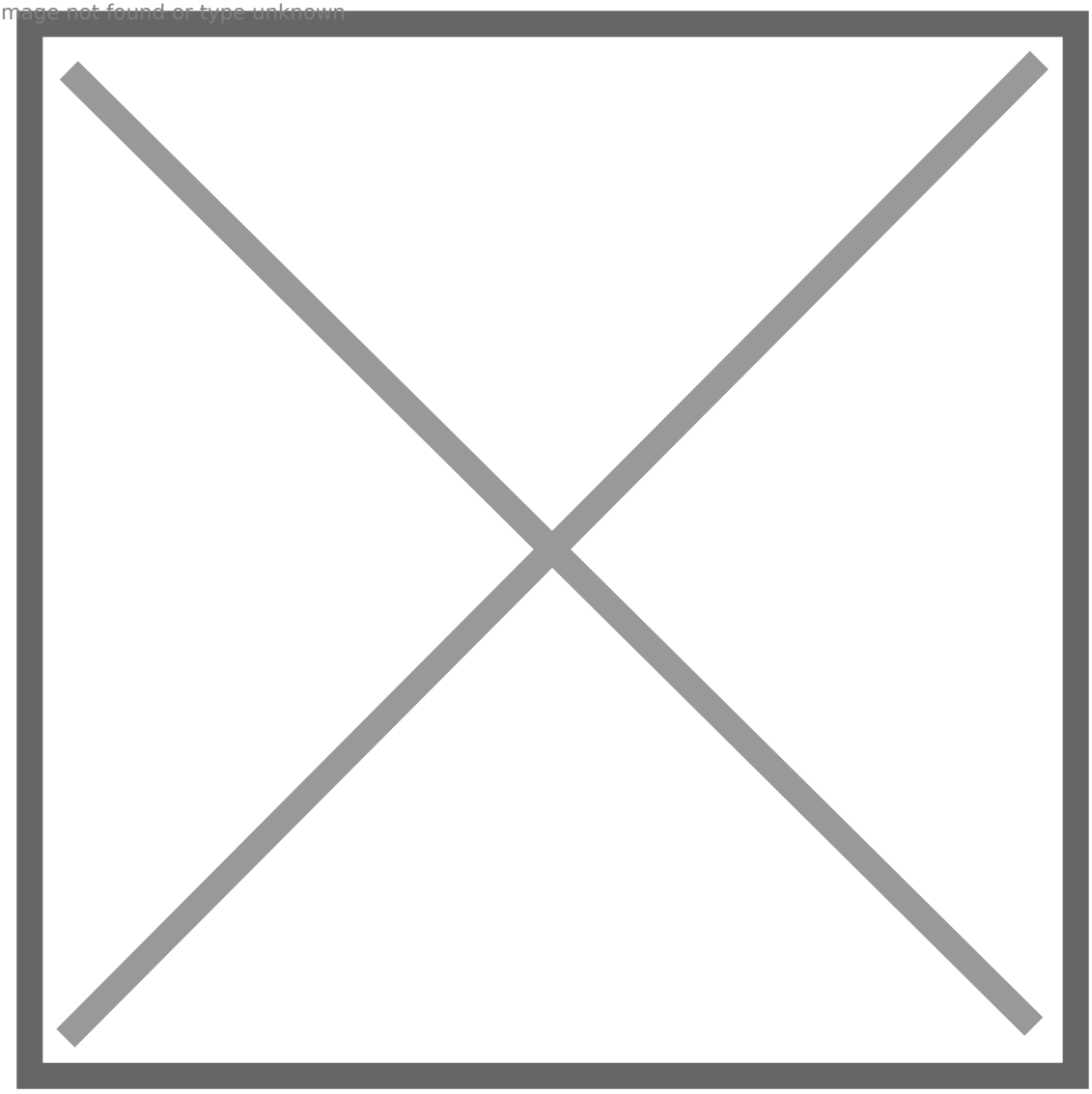


Image not found or type unknown



We can clearly see that the credential is helen:Summer2022!. Since helen.park is a domain user in BLACKOPS.LOCAL, so credential reuse is possible, we should be aware of that. So we completed configurations of file01.

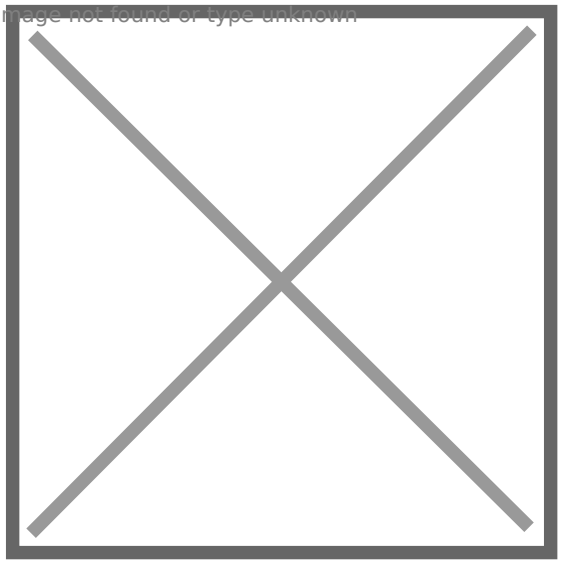
CLIENT SERVER

client01.blackops.local

Now we successfully configured all Linux domain computers. Let’s configure the client server client01.

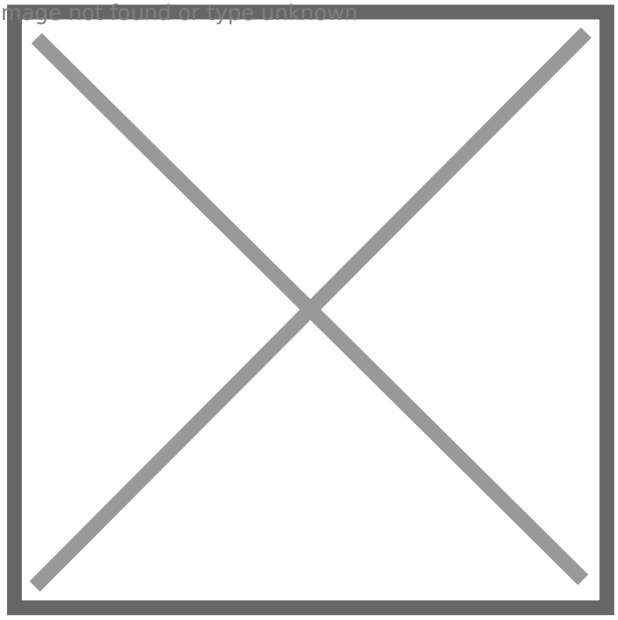
The first step is still configuring DNS. But we also need to disable IPv6.

Image not found or type unknown



And set DC as DNS server.

Image not found or type unknown



We do not need to configure any app or services on client01, but some common settings on Windows hosts.

AppLocker

Run Local Group Policy Editor, enable DLL rules, and enforce all types of rules.

Image not found or type unknown

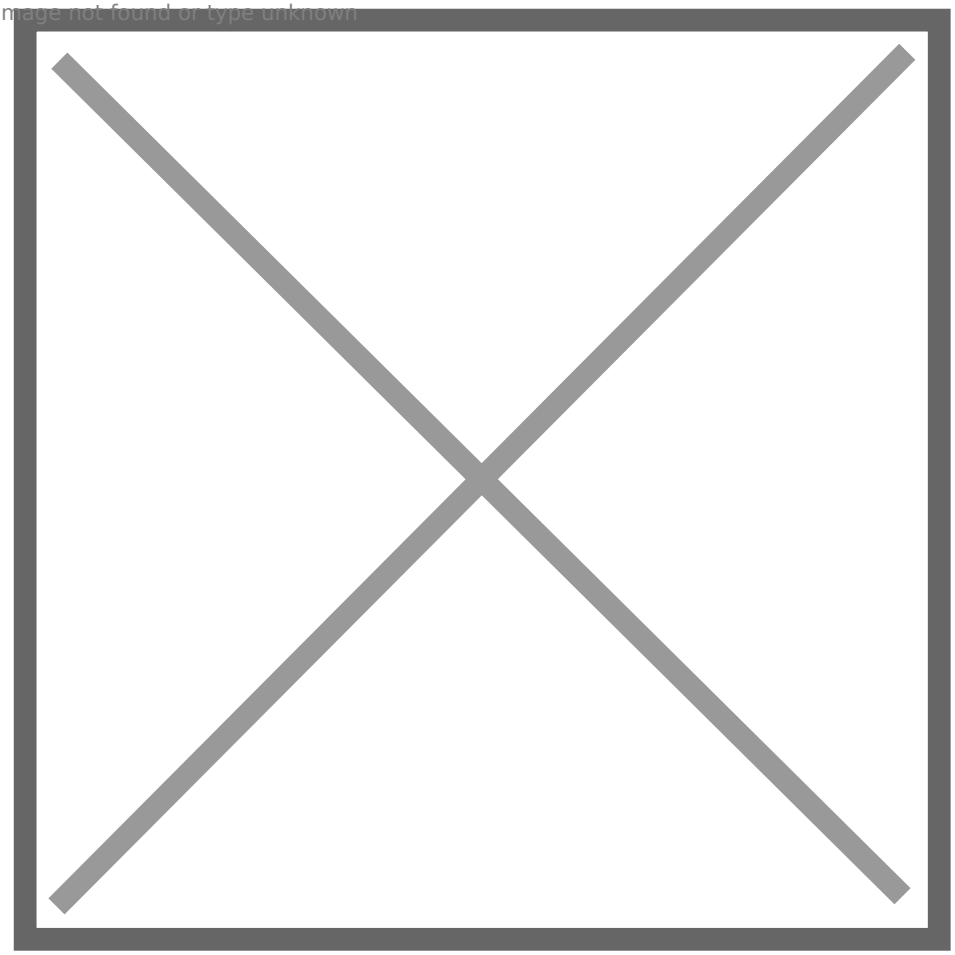
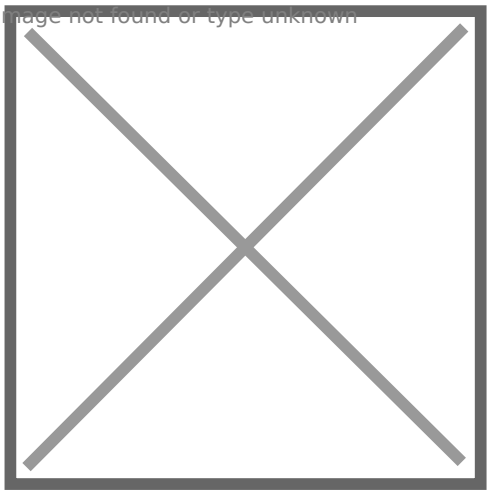
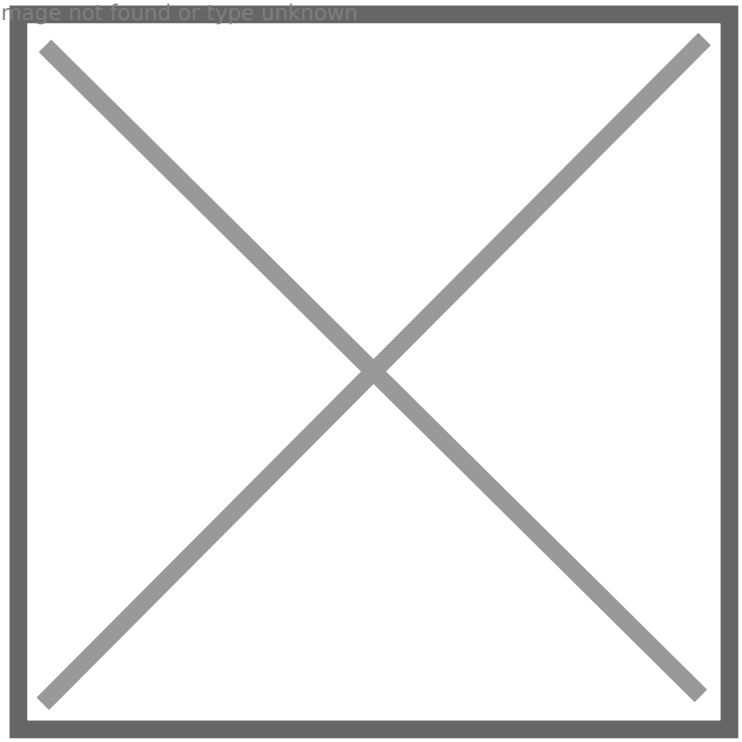


Image not found or type unknown



Enforcing default rules is okay, even though some paths can be abused to execute binary such as C:\windows\tasks. So it is not necessary to create a custom tradecraft or download bypass-clm (https://github.com/calebstewart/bypass-clm) from github.

Image not found or type unknown



Windows Defender

Just use the default settings.

Firewall

I turn off firewall on all windows machines. You could turn on it if you would like to increase a little more difficulty: D

Autologin

Set autologin for domain user helen.park.

UAC

I don’t think UAC bypass is needed in the whole exploitation process, so just leave it default.

Remote Desktop

Enable Remote Desktop setting, and add helen.park to localgroup Remote Desktop Users: **net localgroup “Remote Desktop Users” helen.park /add**

But just as I previously said, we can also achieve this by linking and enforcing a GPO.

Image not found or type unknown

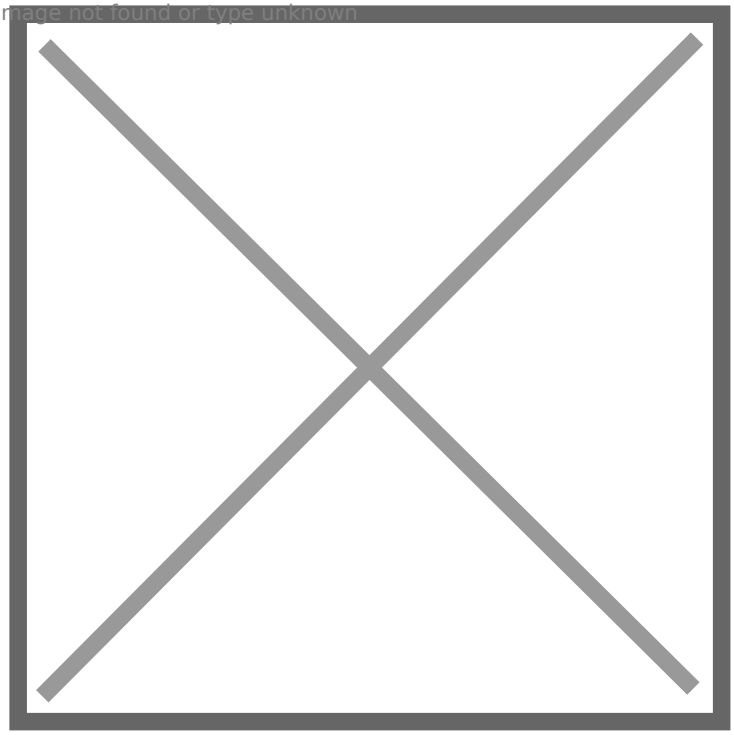
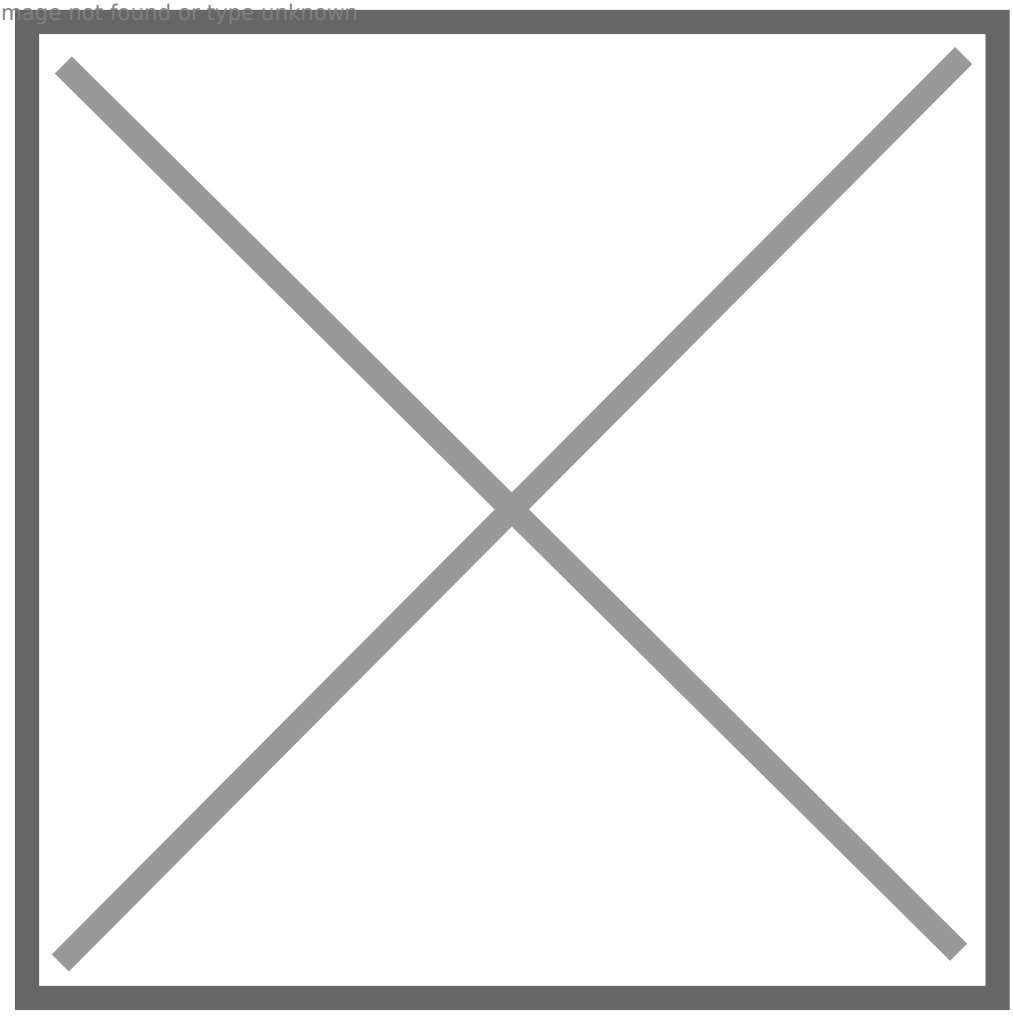


Image not found or type unknown



Then we need to create a script to connect to file01’s FTP server as helen, and two txt file as well.

txt file 1: **FTP Auth.txt**

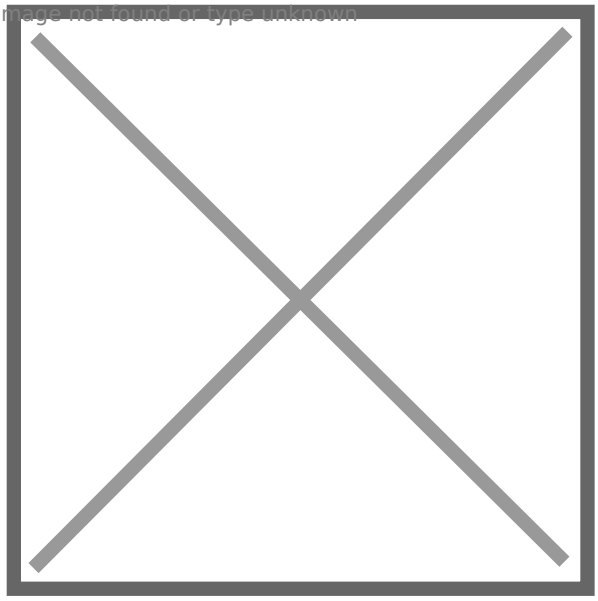
open 192.168.0.52

USER helen

Summer2022!

bye

Image not found or type unknown



script: **script.ps1**

Do {

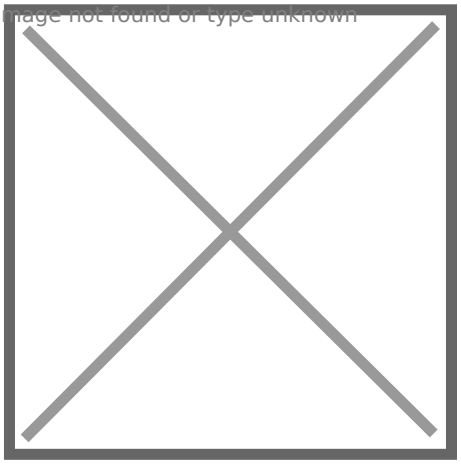
ftp -v -n -s:’.\FTP Auth.txt’

start-sleep -s 3

}

while (1 -ne 2)

Image not found or type unknown



Put these 2 files on helen’s document folder.

Create a **scheduled task** to invoke **powershell.exe** to run the script at logon, so we do not need to manually run it every time we boot client01.

Then create another txt file on helen’s desktop: Resolved Ticket.txt

After finishing editing, just delete it. I just want people not to forget to check Recycle Bin during enumeration.

Image not found or type unknown

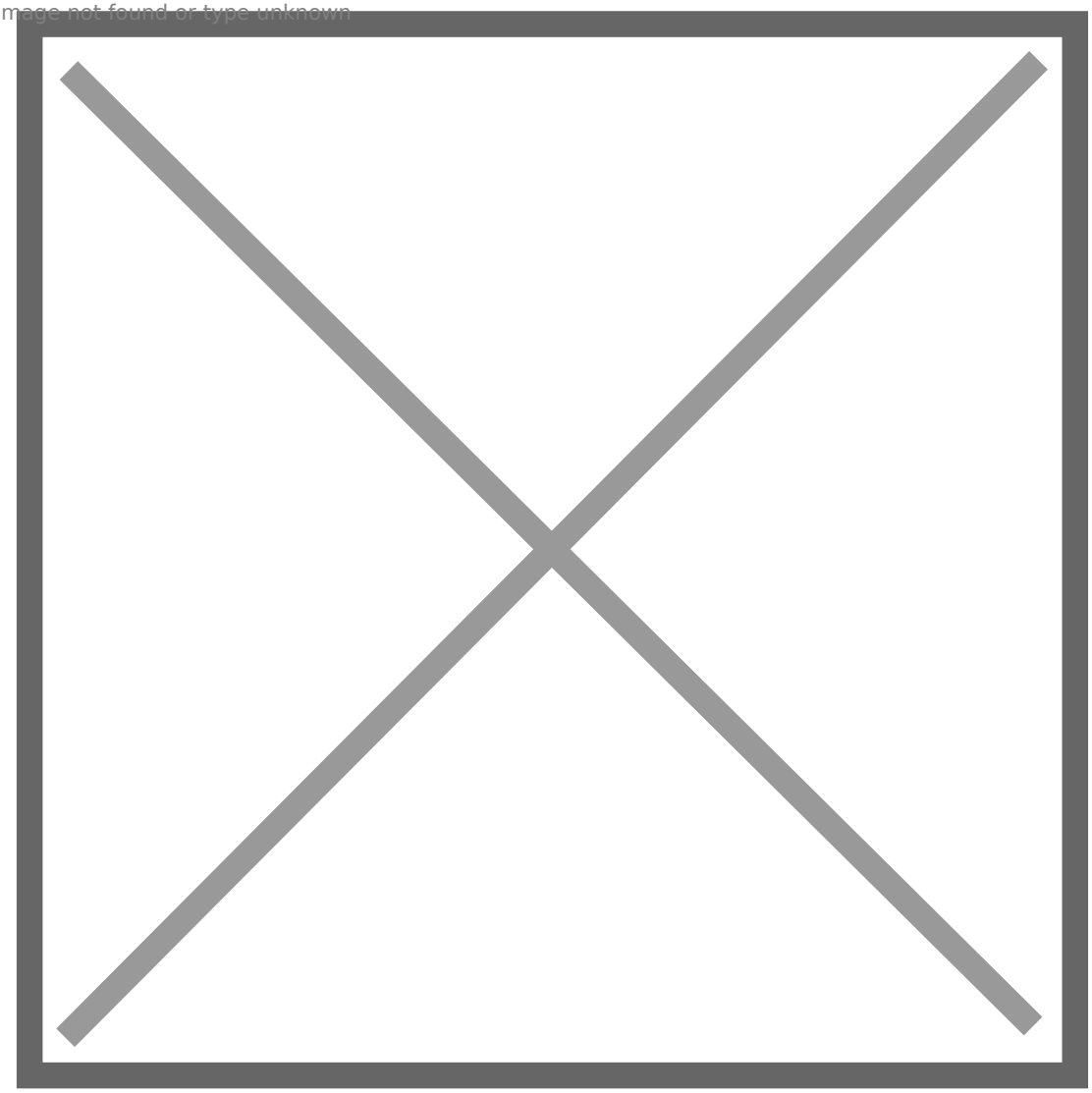
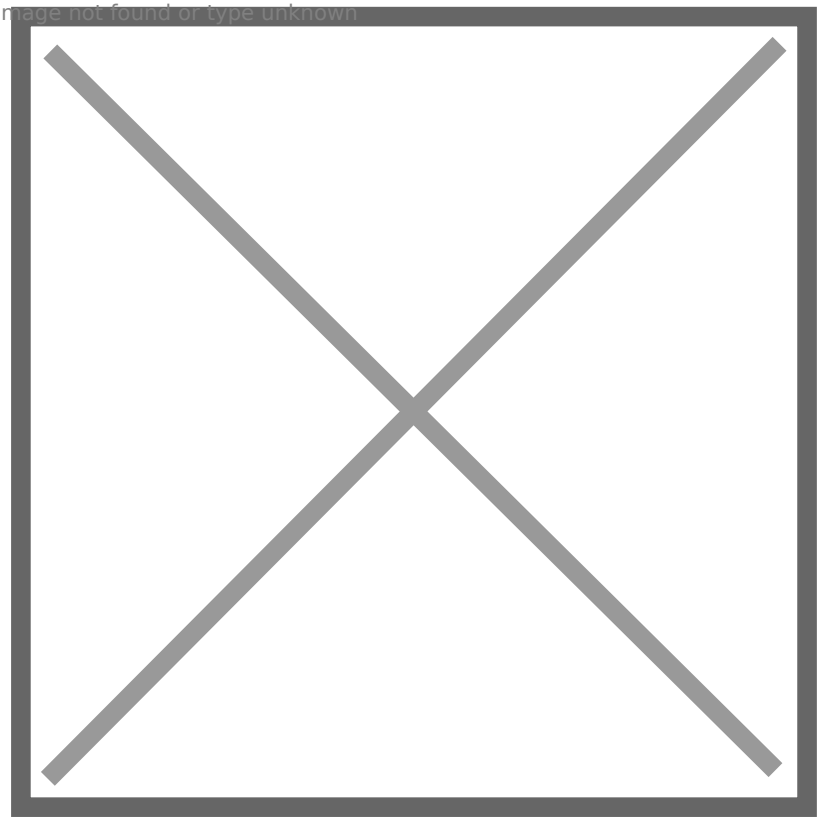


Image not found or type unknown



So we complete the configurations here. Let’s move to SRV01.

SERVER 1

srv01.blackops.local

So we move to the most difficult part of design and configurations. Fortunately, most steps are the same for both SRV01 and SRV02.

First, disable IPv6, then configure IP and DNS.

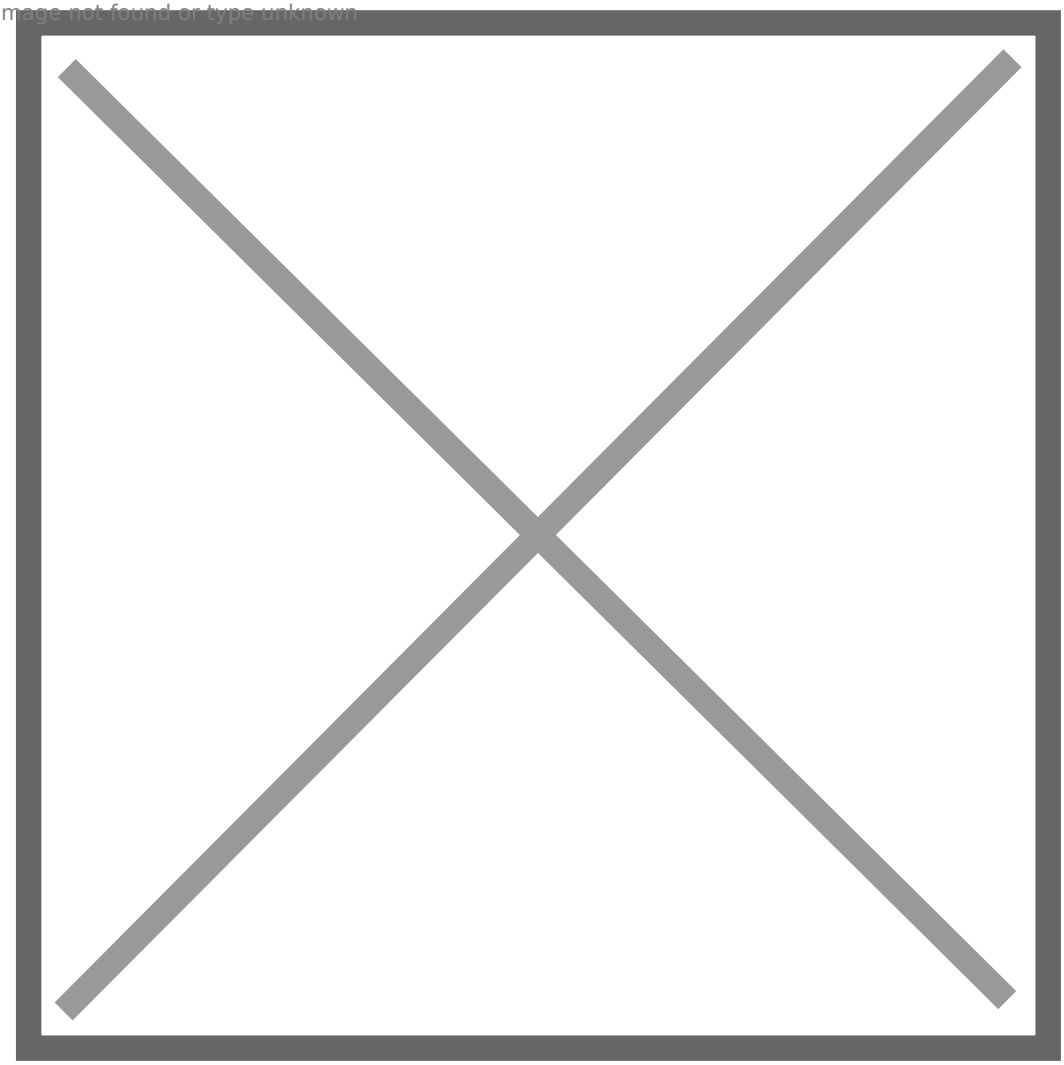
Configure autologin for **jason.hudson**.

Configure AppLocker just as we did on client01.

Add jason.hudson to local group “Remote Management Users” and “Remote Desktop Users”: **net localgroup “Remote Management Users” jason.hudson /add && net localgroup “Remote Desktop Users” jason.hudson /add**

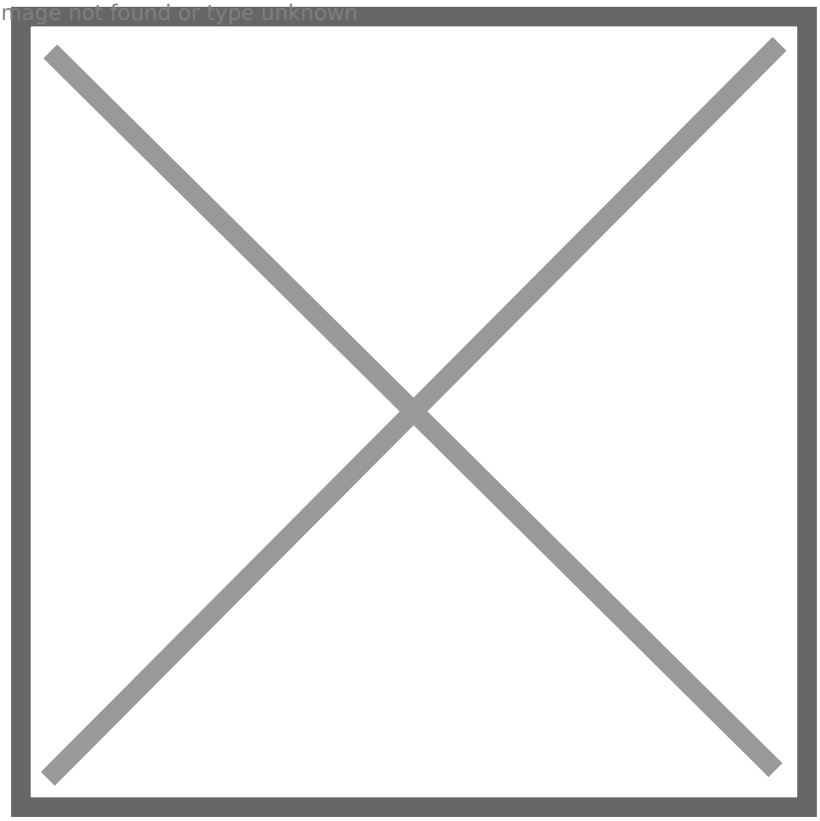
By this way, jason.hudson has WinRM access to SRV01.

Image not found or type unknown



Beside, let’s configure a privilege escalation vector: **AlwaysInstallElevated**. First, we need to add reg key for HKLM: **HKLM\SOFTWARE\Policies\Microsoft\Windows\Installer**, and HKCU: **HKCU\SOFTWARE\Policies\Microsoft\Windows\Installer.** Then add a value for both of them: **AlwaysInstallElevated**, **DWORD** value **1**.

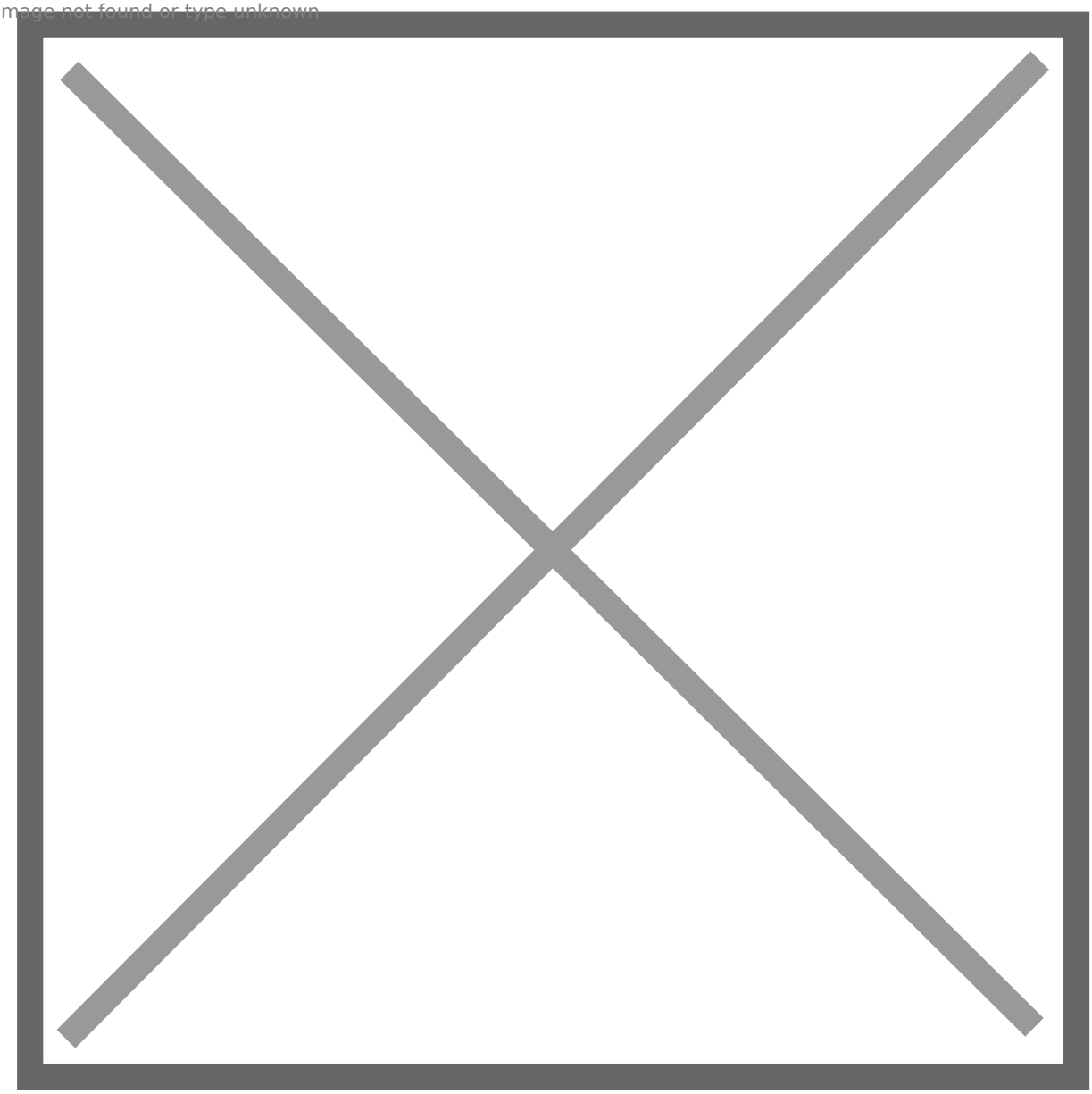
Image not found or type unknown



Besides, we also need to add this value for jason.hudson domain user, since HKCU seems to have no effect on domain users.

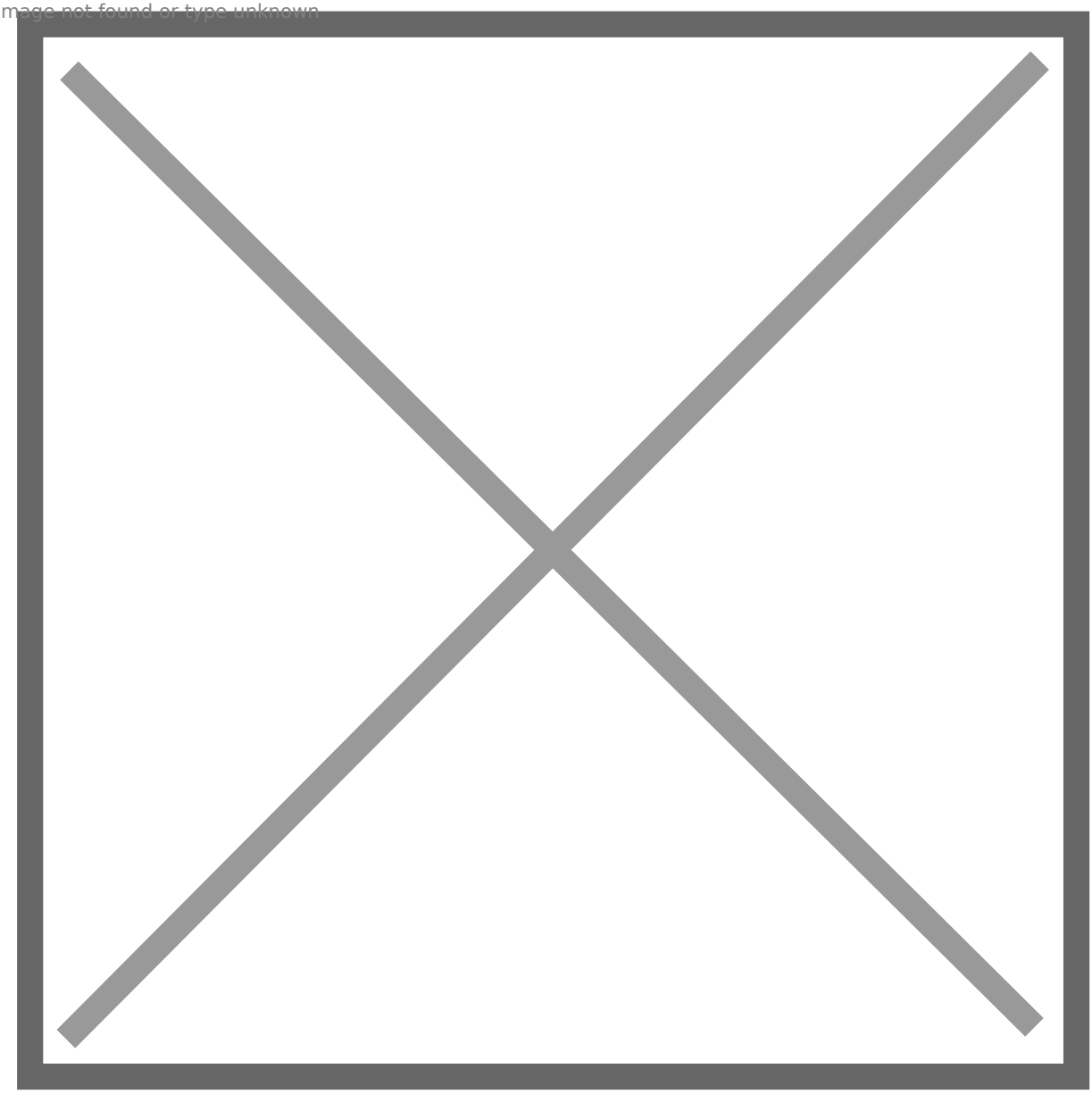
Unfold **HKEY_USERS**, find jason.hudson’s **SID**, and add this key-value as previous.

Image not found or type unknown



Besides, we need to configure Local Group Policy Editor: **Computer Configuration > Administrative Templates** > Windows Components > Windows Installer, edit “**Turn off Windows Installer**”, change the setting as following screenshot.

Image not found or type unknown



Otherwise the normal user cannot install a package.

Enable **PPL** for SRV01, just follow the steps in the link: https://docs.microsoft.com/en-us/windows-server/security/credentials-protection-and-management/configuring-additional-lsa-protection

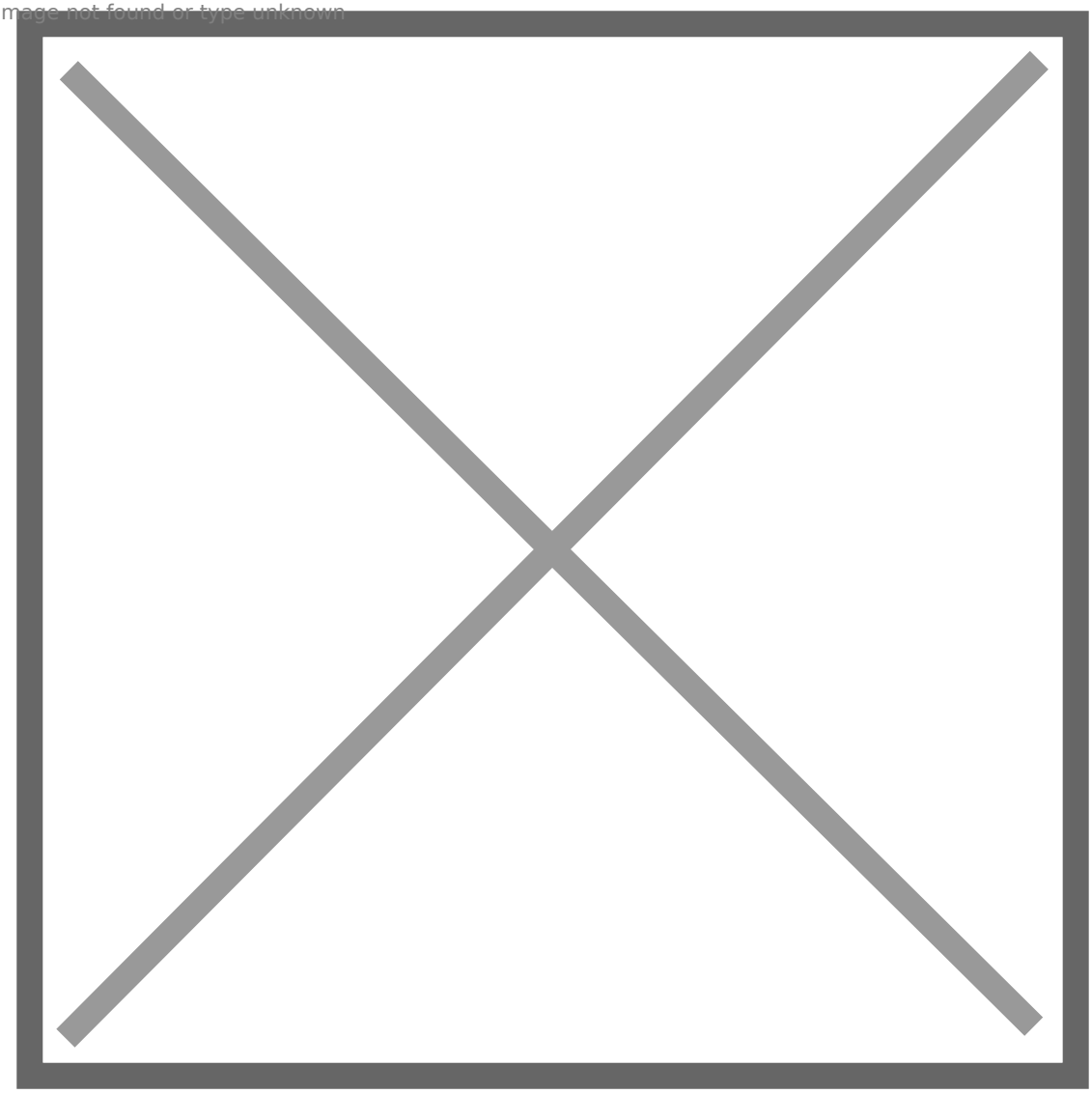
Now let’s install and configure SQL Server 2019, it is the most difficult and complex part.

Download link: https://www.microsoft.com/en-us/sql-server/sql-server-downloads (Developer)

SSMS: https://docs.microsoft.com/en-us/sql/ssms/download-sql-server-management-studio-ssms?view=sql-server-ver16

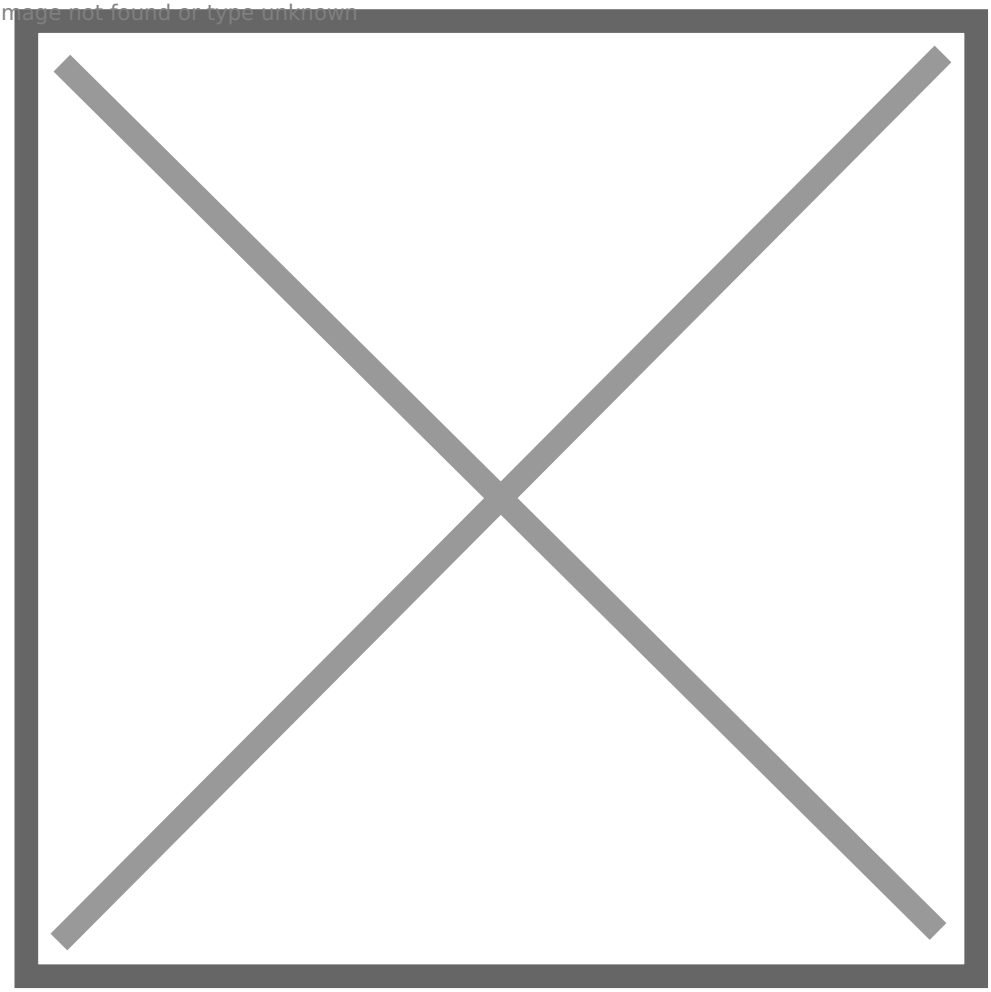
Install Windows SQL Server 2019 first, something important is that choose Customize Installation, because Basic Installation cannot meet our requirements.

Image not found or type unknown



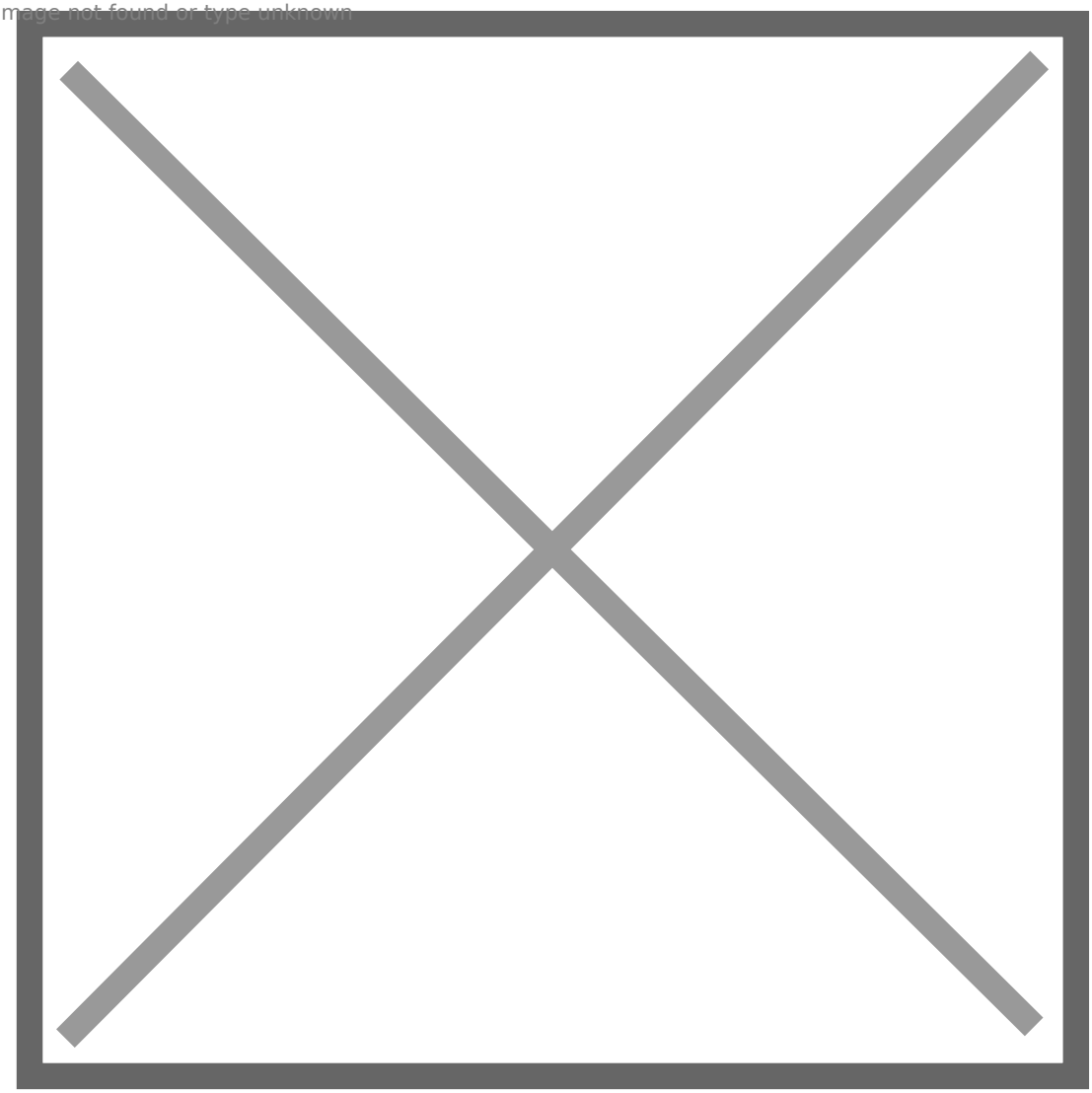
During installation, we can leave most pages default, but something needs customization. When selecting Feature, I cannot tell the minimum selections to make the AD set works, but my selections work well.

Image not found or type unknown



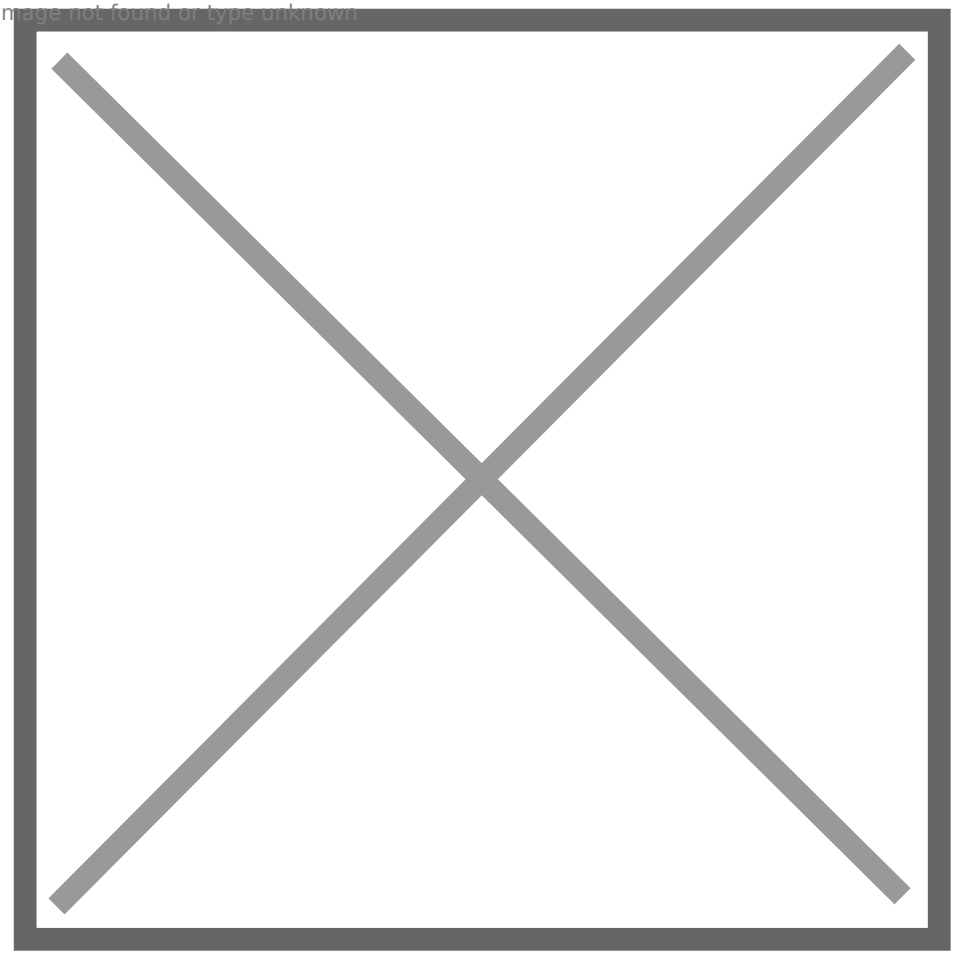
Then you need to specify instance name, to make it in line with my settings, you can change instance name to DB01.

Image not found or type unknown



Next, leave service accounts default. When configuring Authentication Mode, choose Mixed Mode. After that, it is recommended to click Add Current User button to add current local admin to sysadmin. By this way, both sa and local admin account have sysadmin privilege.

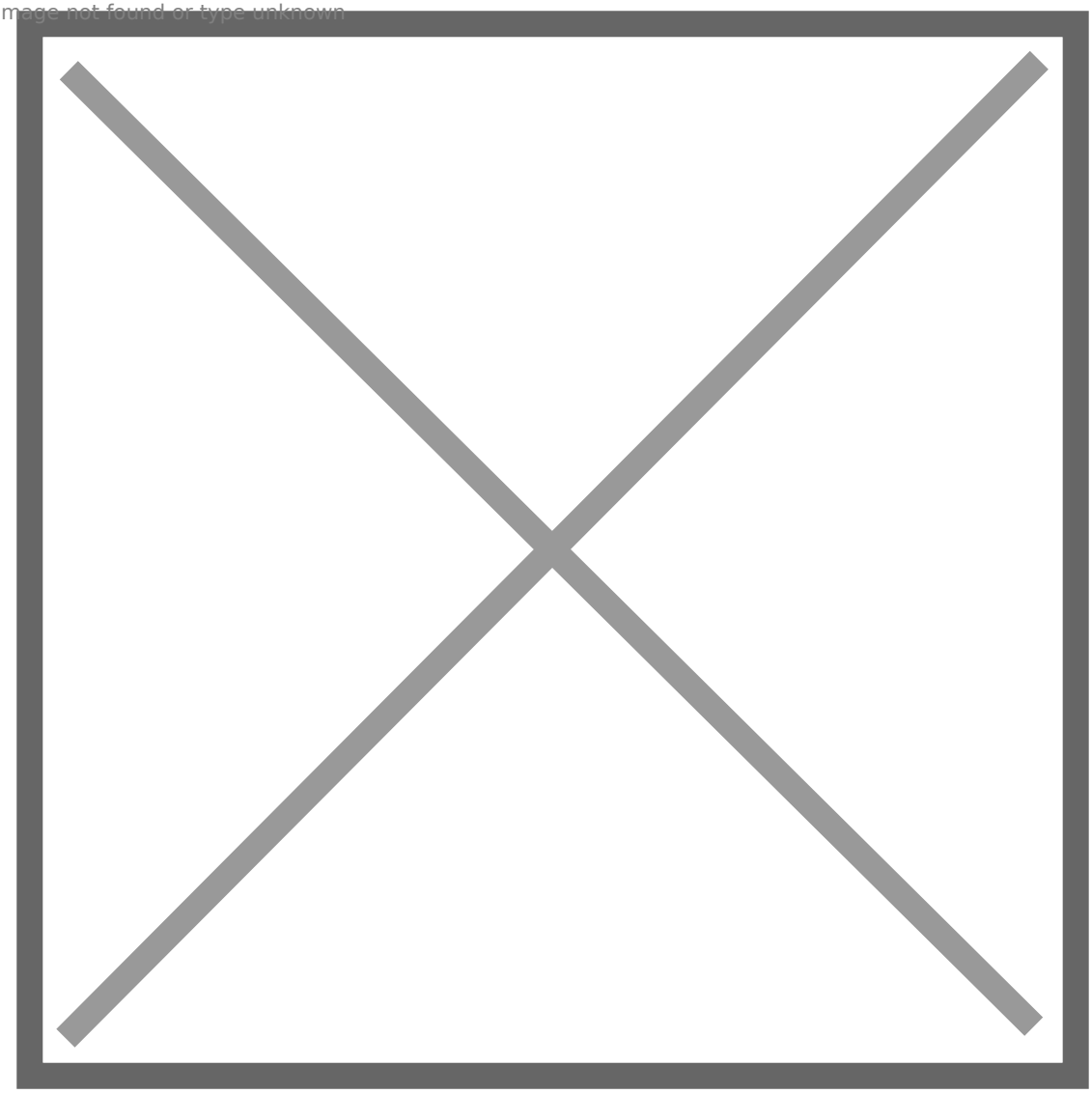
Image not found or type unknown



After setting this, we can leave left default and complete the installation. Installing SSMS is simple, we do not need to customize something.

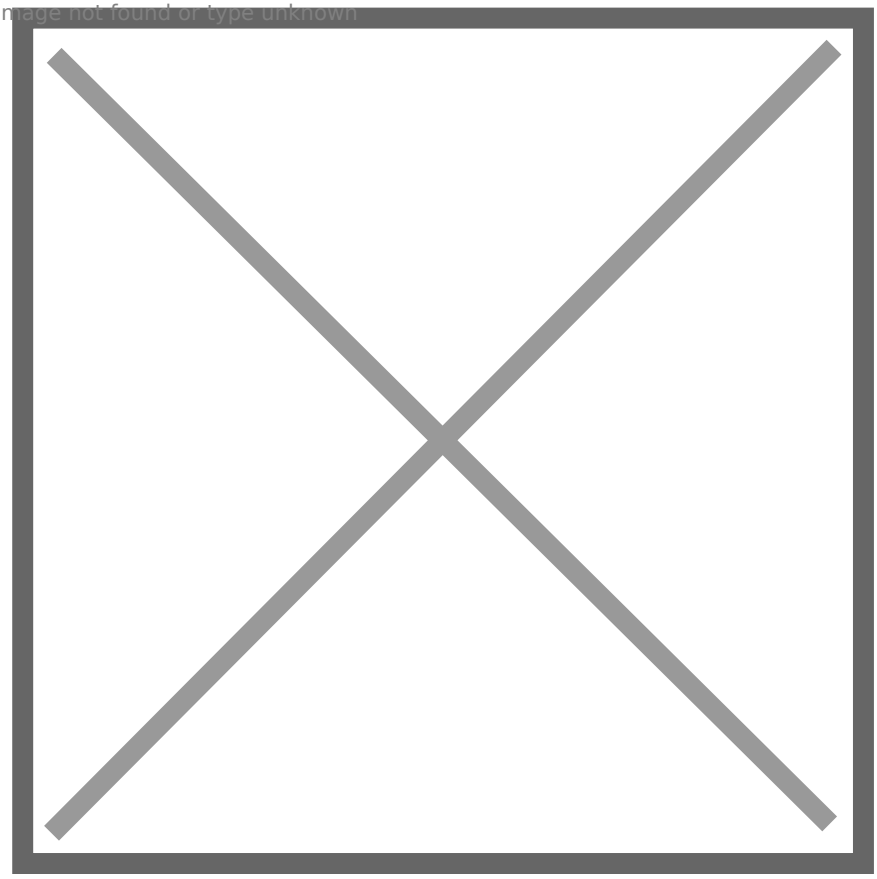
Run **Sql Server Configuration Manager**, we need to modify few settings. First, click **SQL Server Services -> SQL Server (DB01)**, then select **Log On** tab, change logon account to BLACKOPS\svc_sql, type the correct password. Then, we could need a restart of SQL service.

Image not found or type unknown



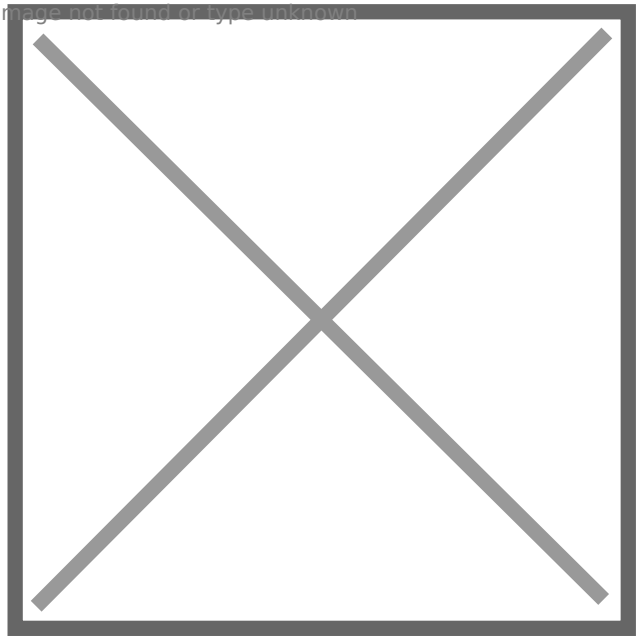
Second, click **SQL Server Network Configuration -> Protocols for DB01 -> TCP/TP**, enable it.

Image not found or type unknown



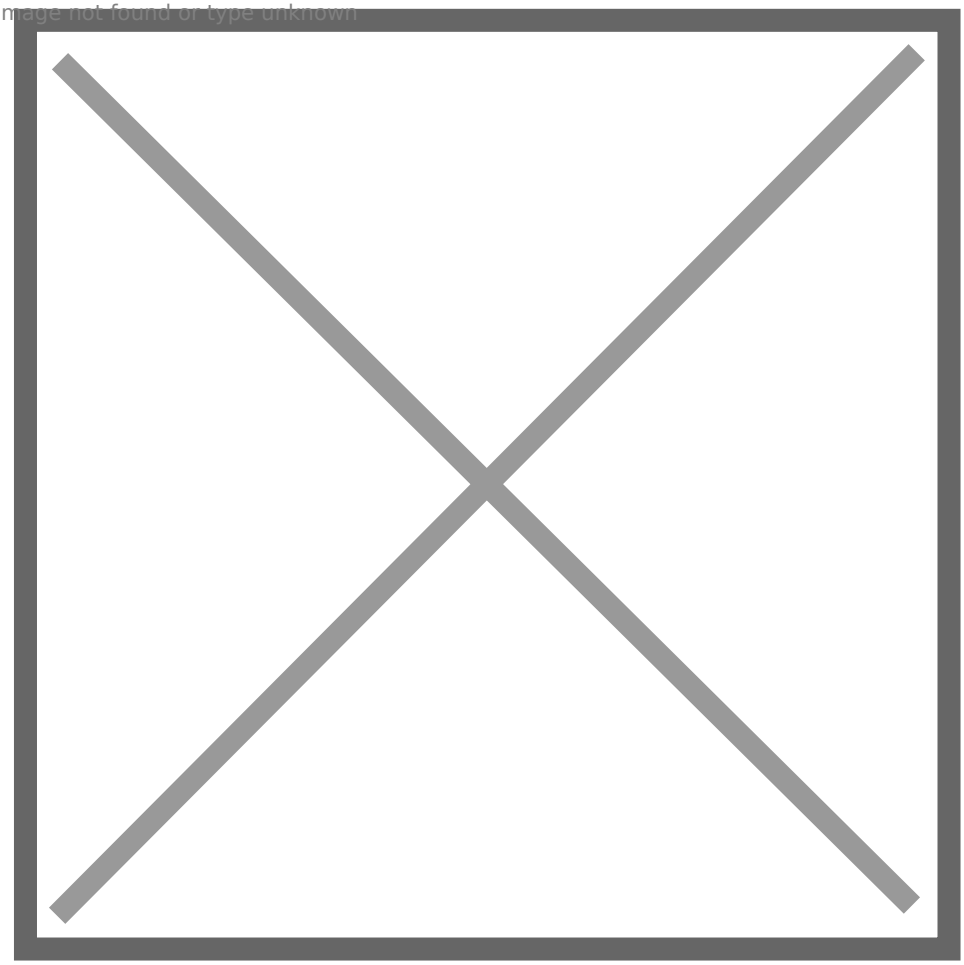
Then double click it, select **IP Address** tab, leave all **TCP Dynamic Ports** blank, and set all **TCP Port** to 1433.

Image not found or type unknown



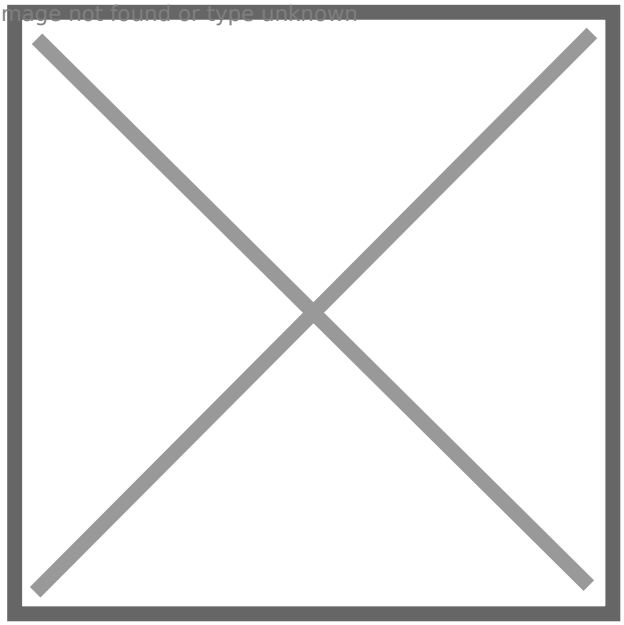
Why we need to disable dynamic ports? Because we will set SPN for svc_sql to make SQL Server supports Kerberos authentication. We also need to set **Start Type** of service **SQL Server Browser** to **Automatic**.

Image not found or type unknown



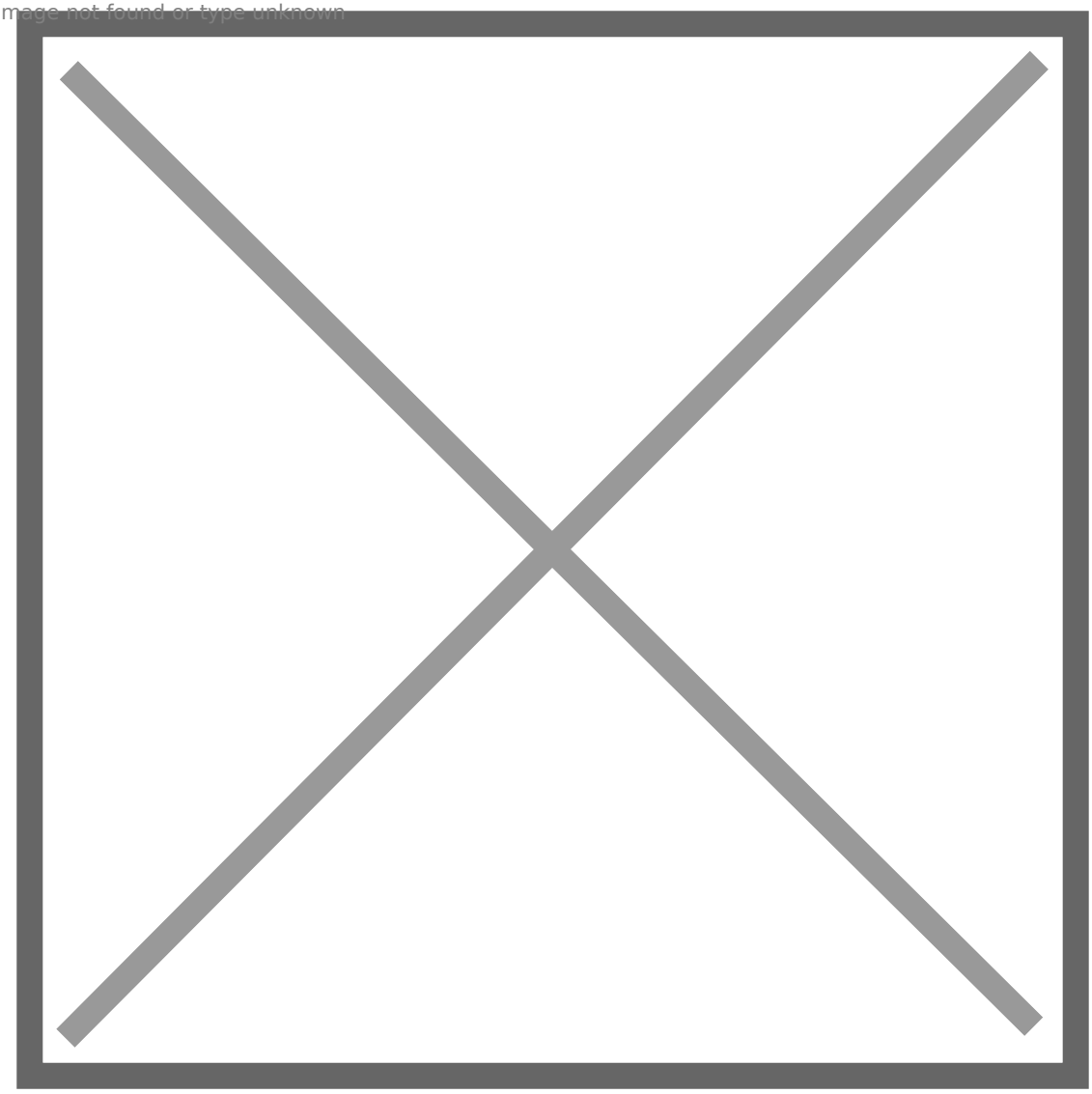
Then, let’s revisit to DC to do some configurations. Run Active Directory Users and Computers, check **Advanced Features**.

Image not found or type unknown



Double click SRV1 (Same steps for SRV02), click **Security** tab and **Advanced** button, add a new permission for svc_sql on SRV1.

Image not found or type unknown



Select principal as svc_sql, apply this permission on this object only. Clear all default check, but check **Read servicePrincipalName**, **Write servicePrincipalName** properties, and **Validated write to service principal name** permission. This official document explains well:

https://docs.microsoft.com/en-us/sql/database-engine/configure-windows/register-a-service-principal-name-for-kerberos-connections?view=sql-server-ver16.

Image not found or type unknown

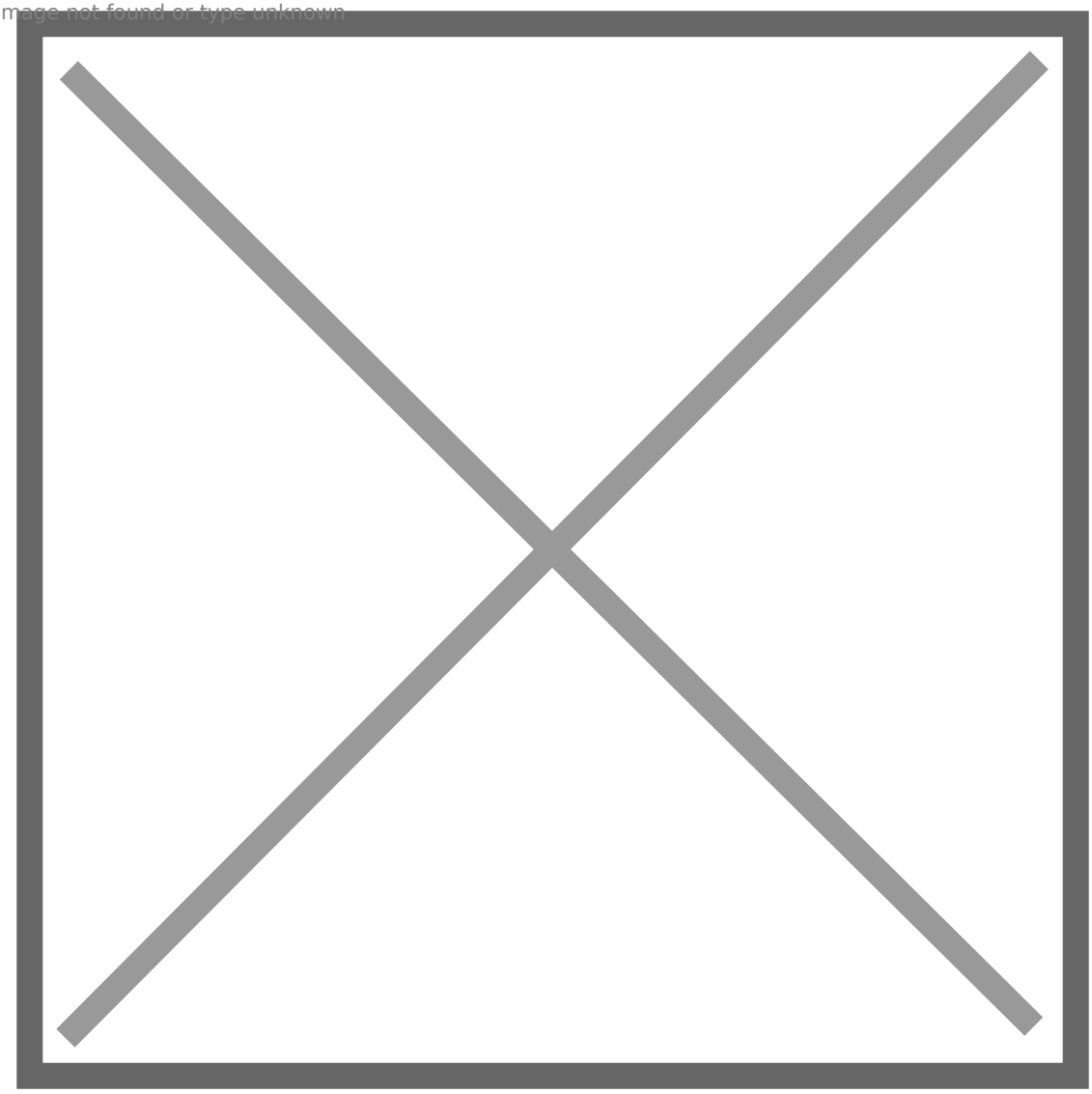
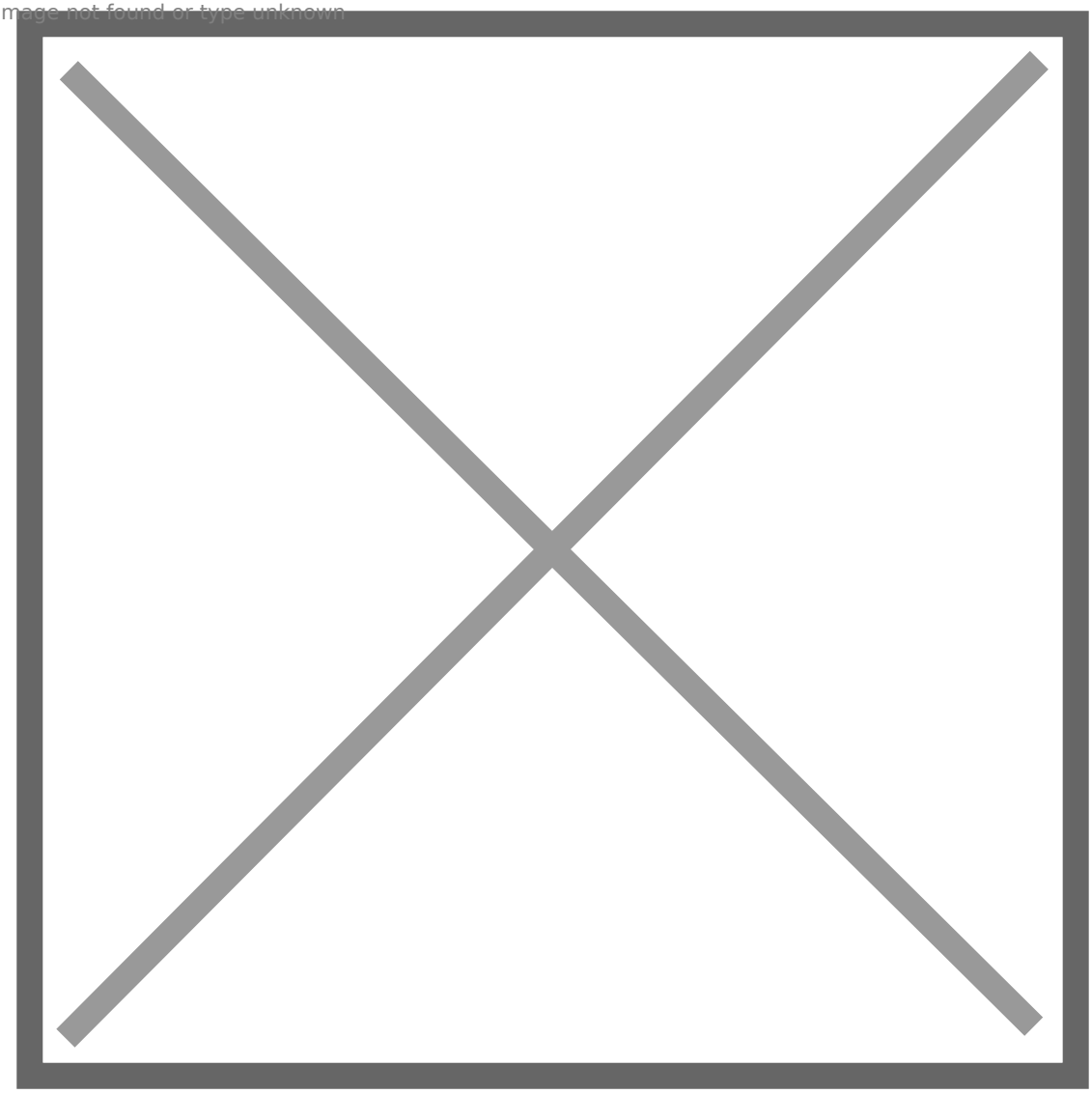


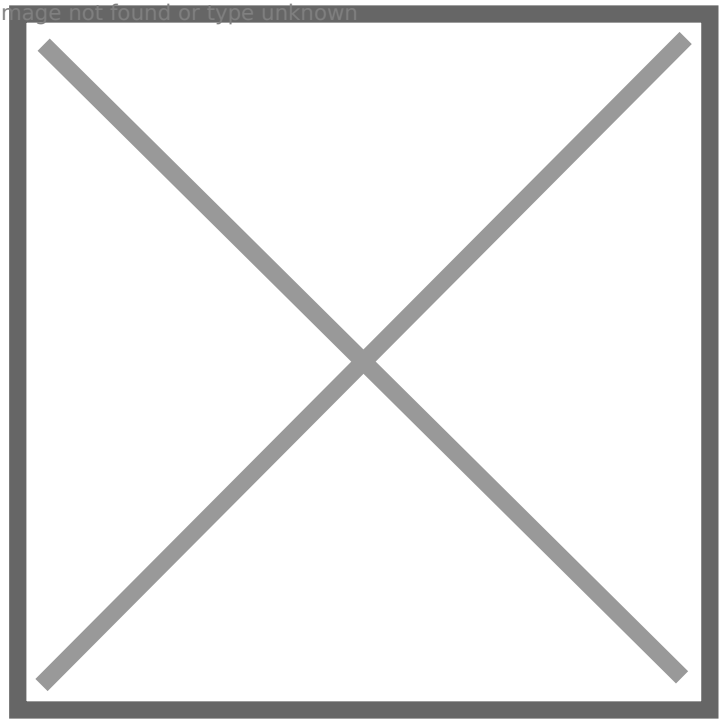
Image not found or type unknown



By the way, let’s configure DACL and delegation.

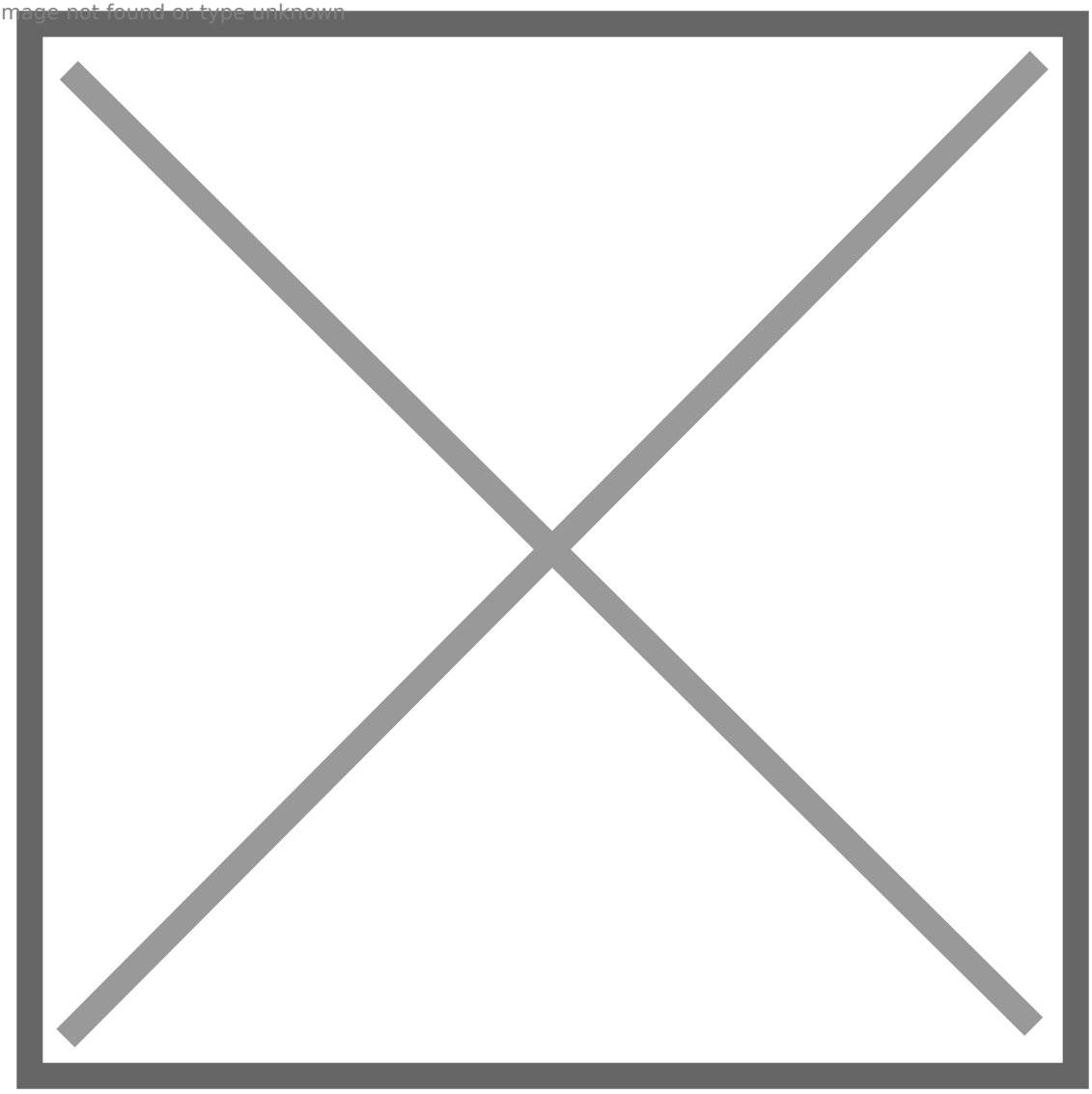
SRV02 is set **unconstrained delegation**.

Image not found or type unknown



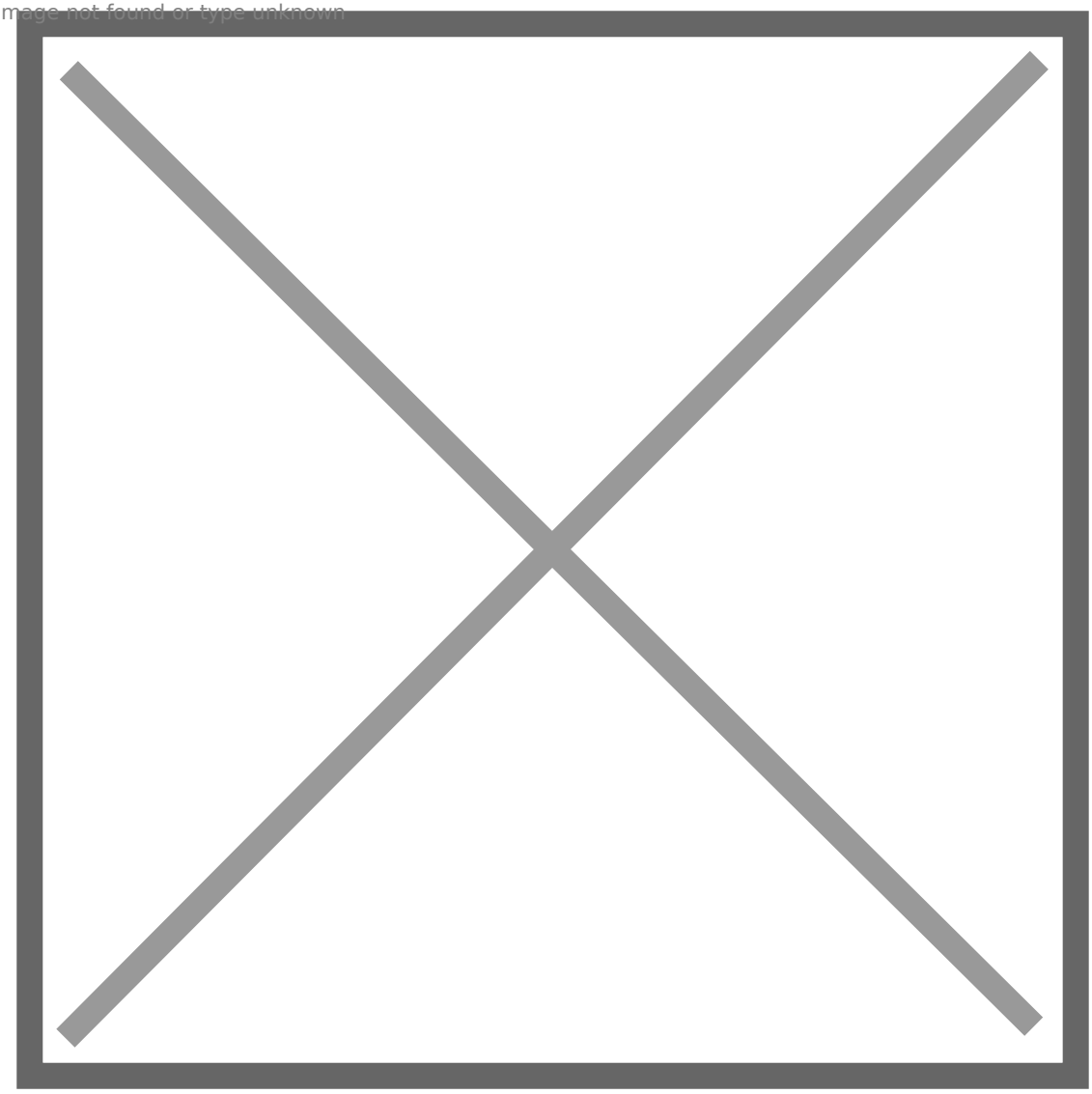
russell.adler has **ForceChangePassword** permission on frank.woods

Image not found or type unknown



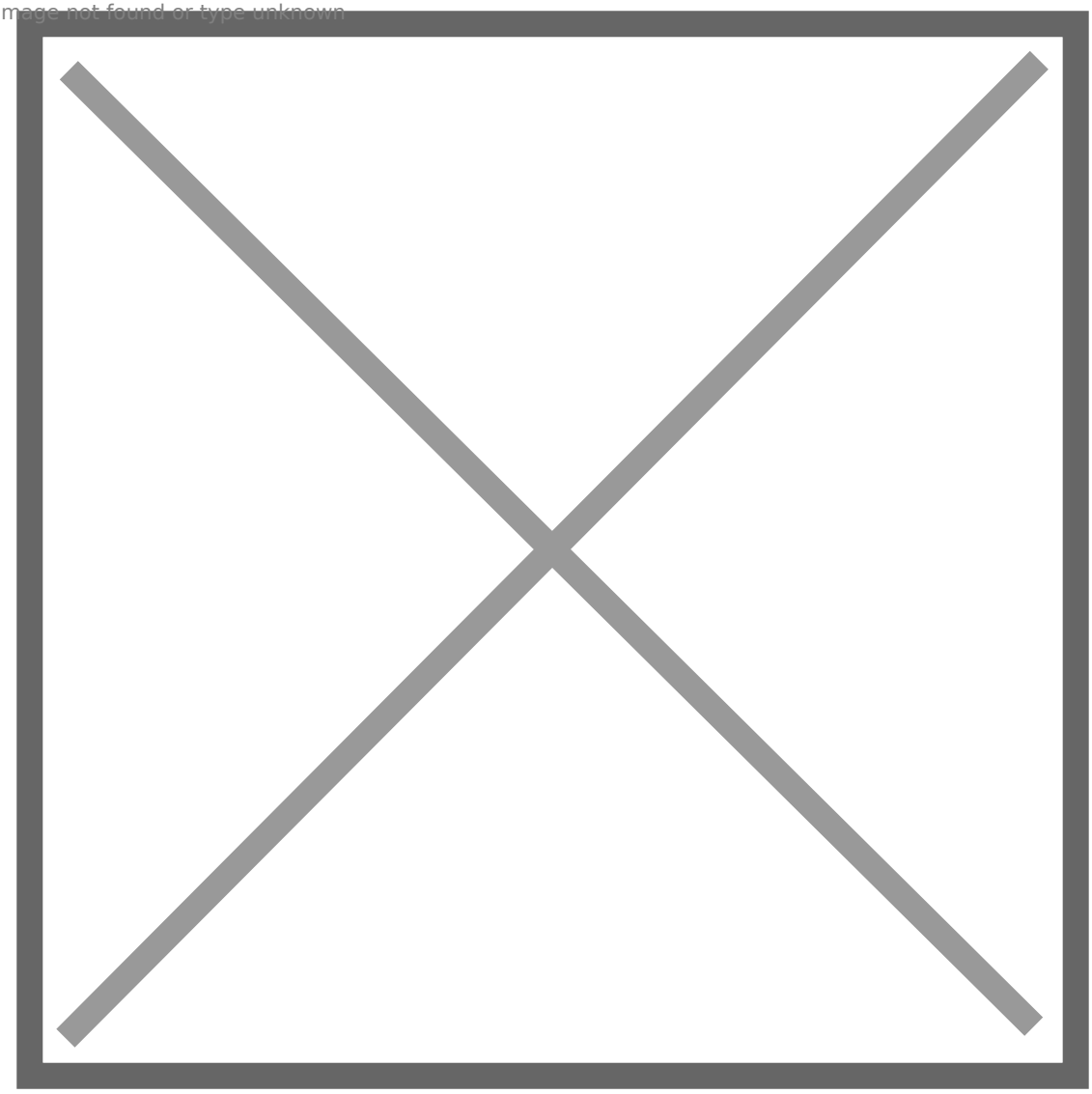
frank.woods has **GeneticWrite** permission on ir_operator

Image not found or type unknown



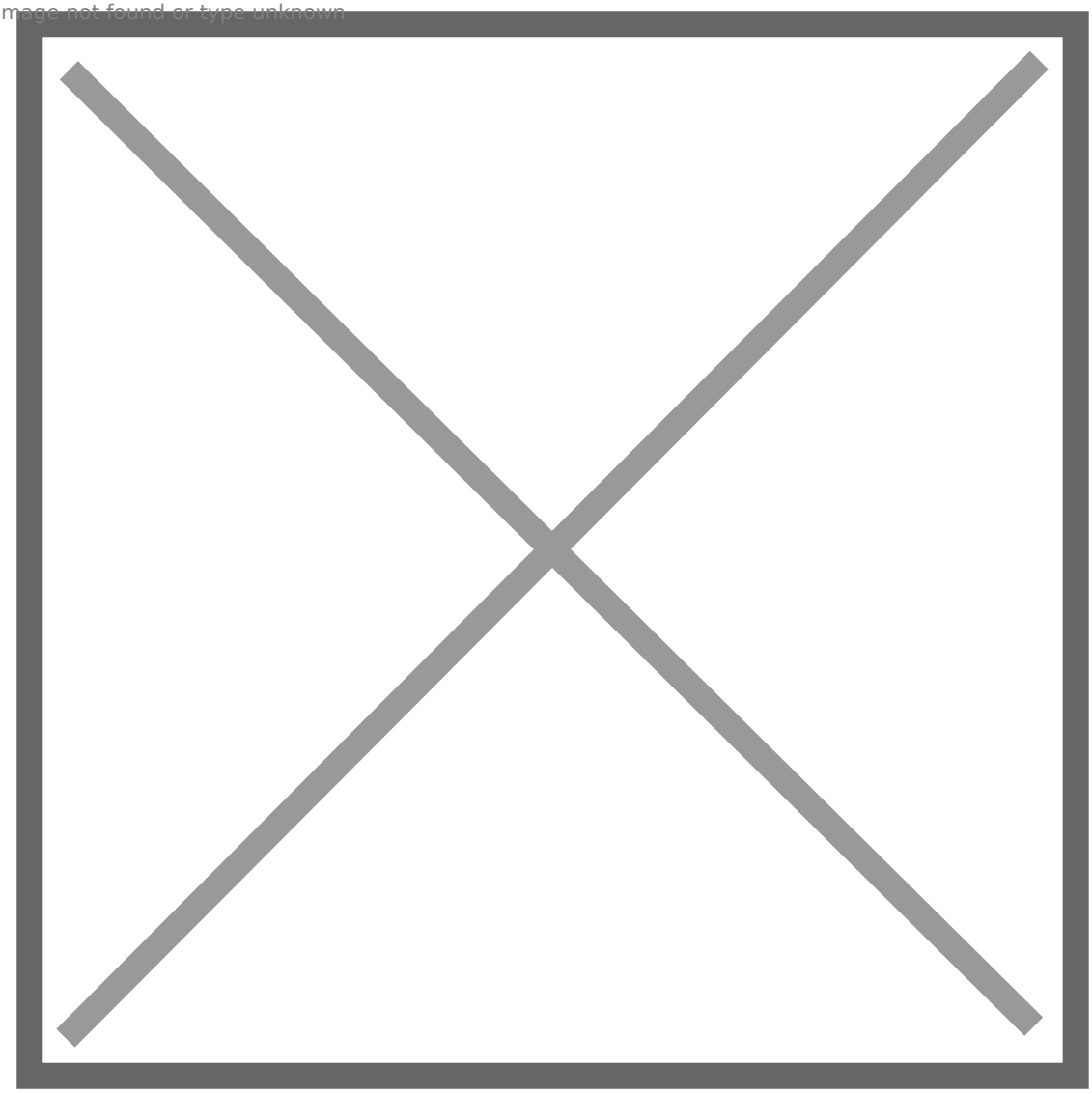
df_operator has **GenericWrite** permission on SRV01

Image not found or type unknown



Cool, all set! Back to SRV01, download a tool from https://www.microsoft.com/en-us/download/details.aspx?id=39046 to help us set SPN automatically. If we did not set proper SPN, it helps us correct it as well. After installing it, run it and connect to the instance, no need to provide any credential. Since we configured proper SPNs, so we do not have to make any change. But if you did not configure SPNs properly, the tool will warn you and you just need to click Fix button.

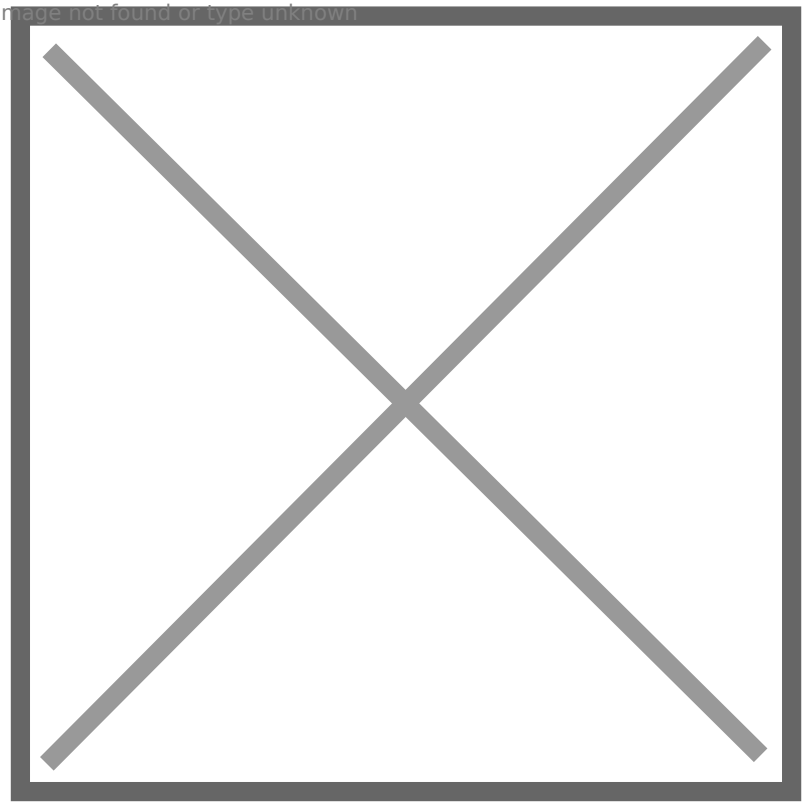
Image not found or type unknown



Now, I believe we successfully set SPN and configure Kerberos authentication for SQL instance. But since the process is complex, I cannot make sure if I miss something. If you follow my steps and cannot reproduce it successfully, please let me know.

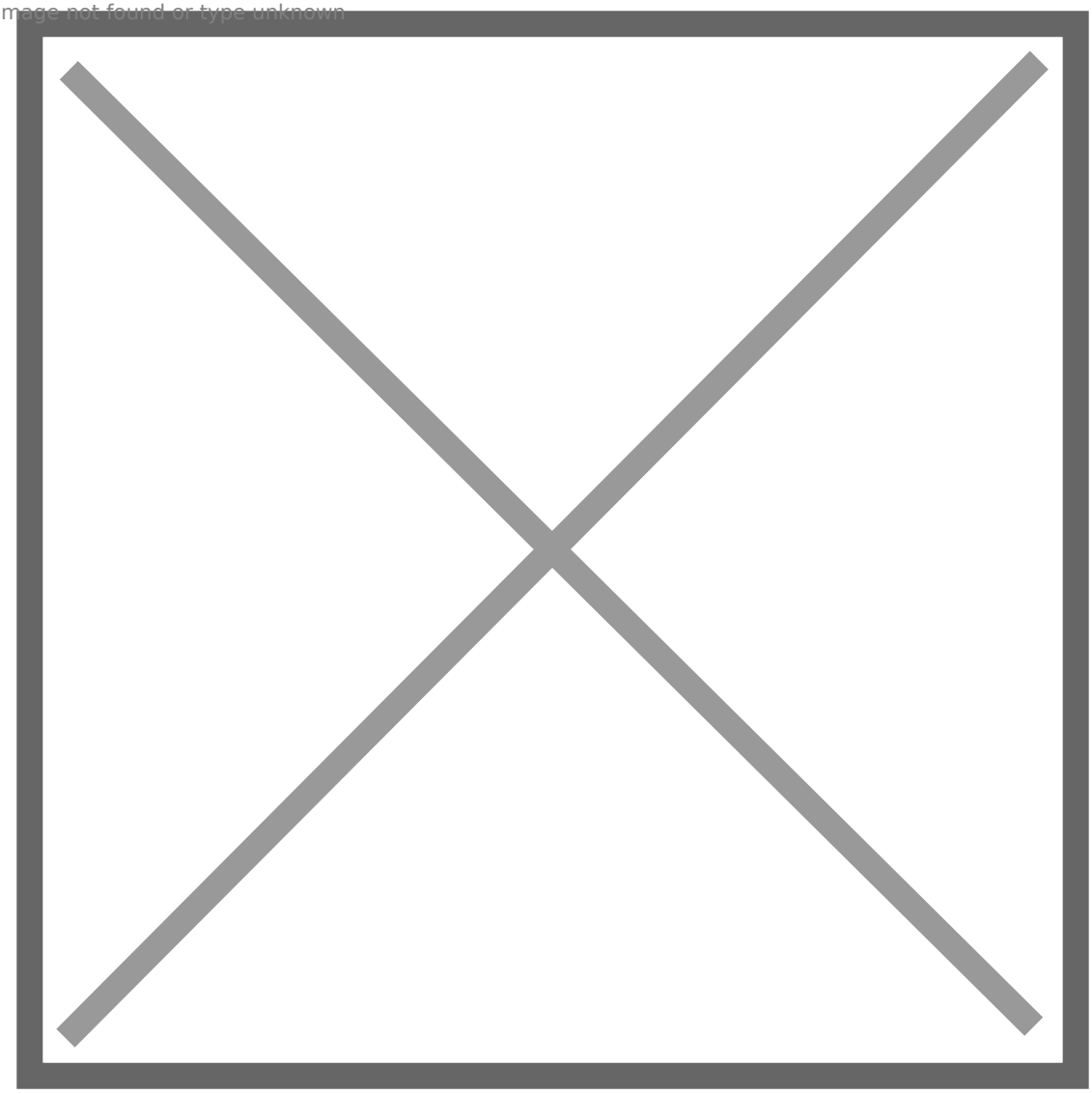
Then, run SSMS 2018, which we installed previously. Change Server name to SRV01\DB01 and select SQL Server Authentication, connect.

Image not found or type unknown



Check SRV01\DB01’s property, make sure **Allow remote connections to this server** is checked.

Image not found or type unknown

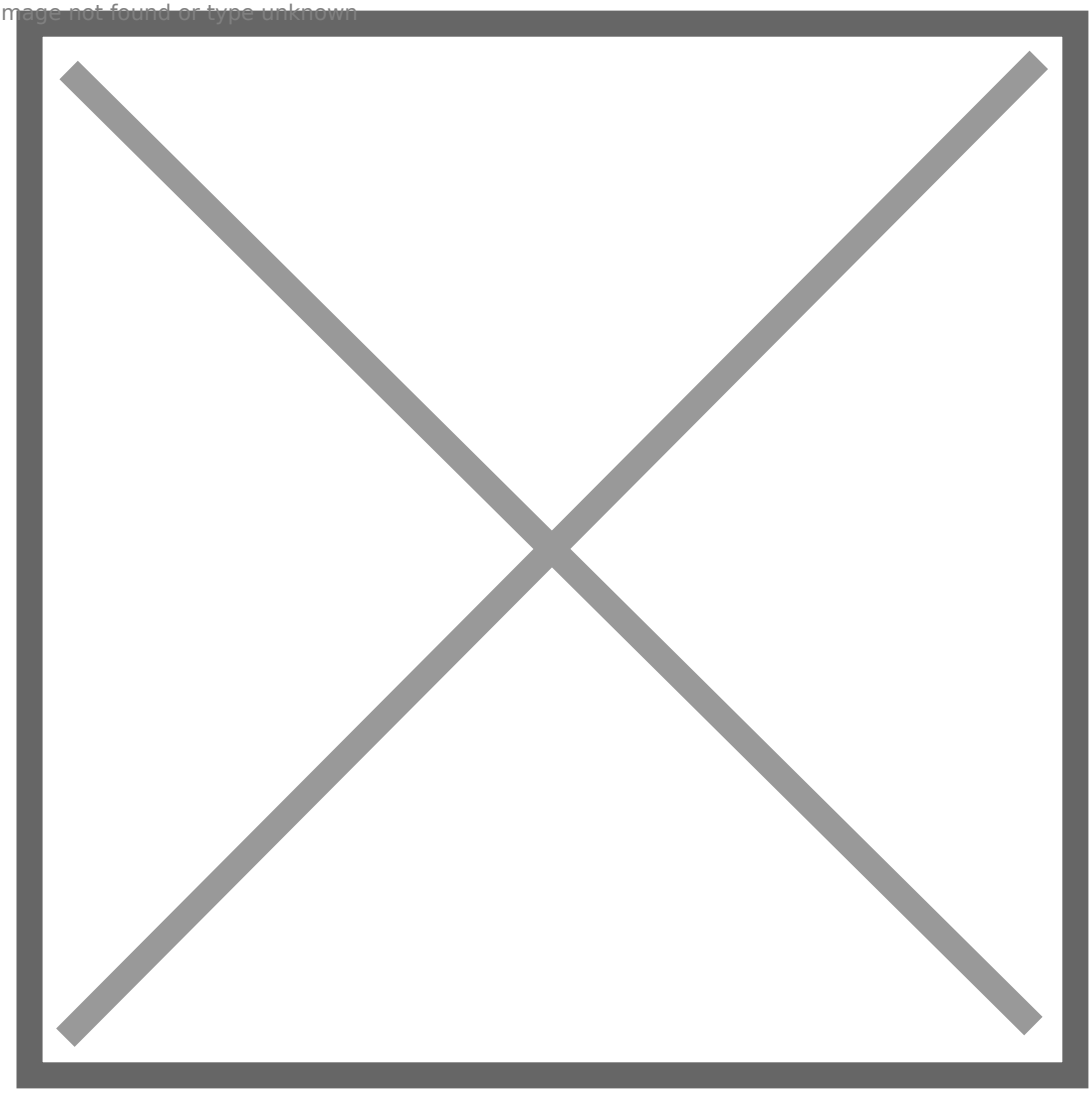


Then, we need to add few logins.

BLACKOPS\Administrator: Sysadmin

Not required, just to make it more realistic.

Image not found or type unknown



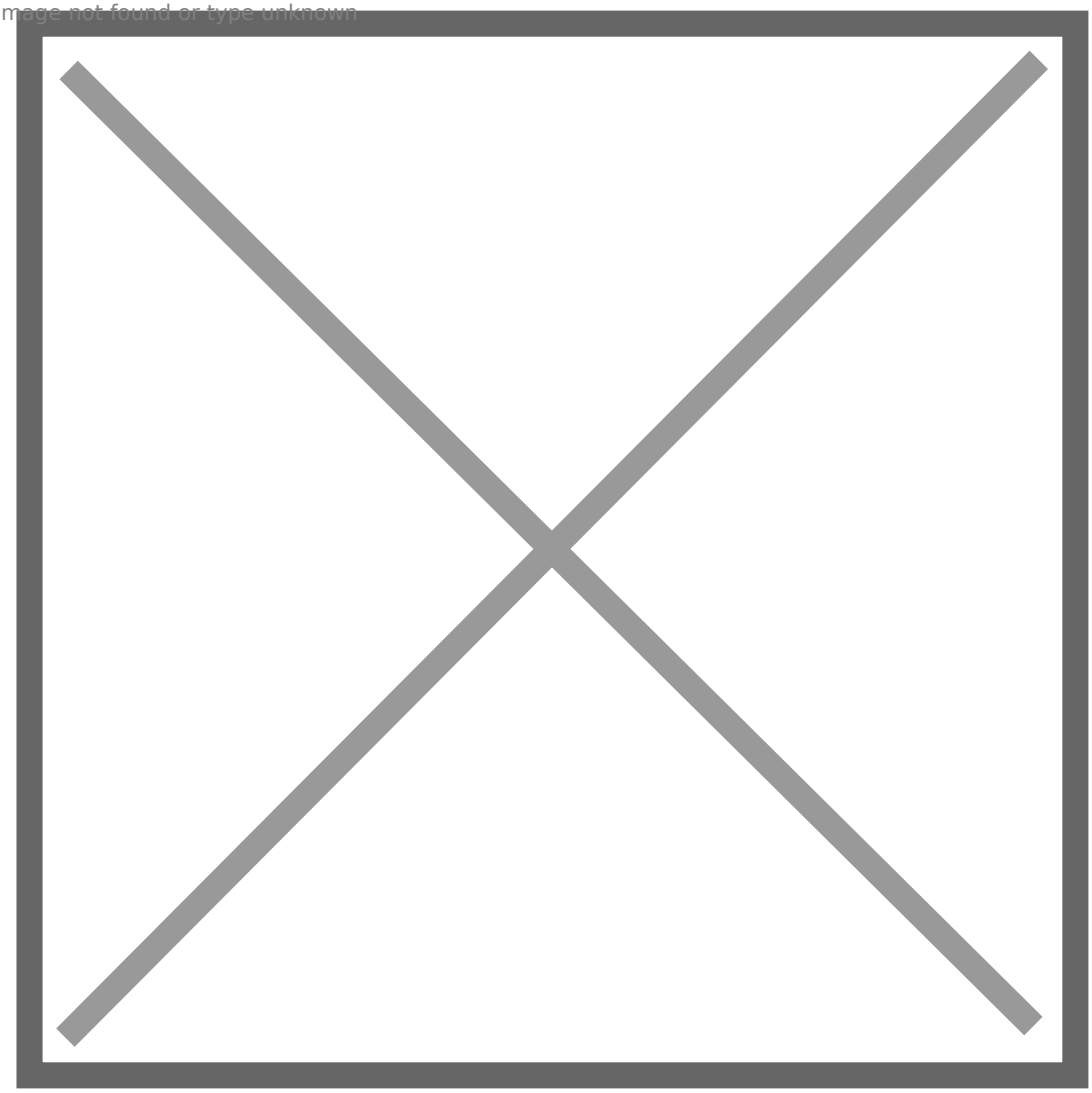
BLACKOPS\Domain Users: Least privilege

Leave everything default

BLACKOPS\svc_sql: Itself is not sysadmin but can impersonate sysadmin.

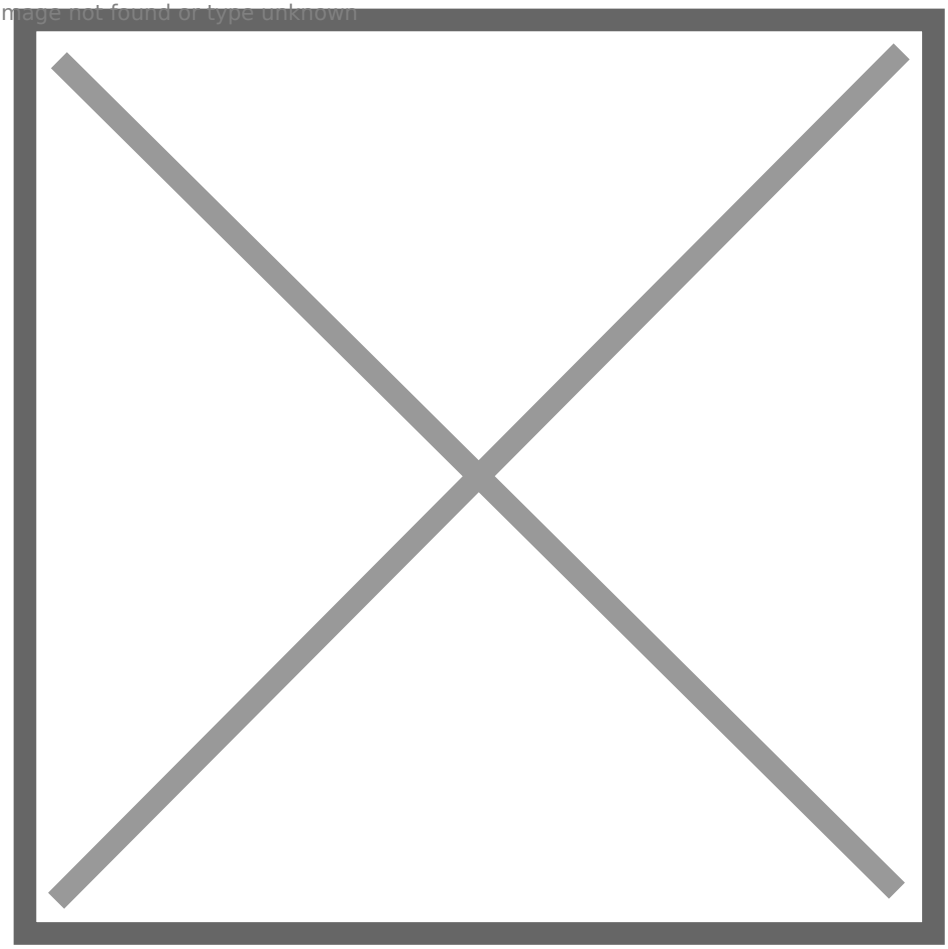
Select few permissions for svc_sql, **IMPERSONATE ANY LOGIN** is required.

Image not found or type unknown



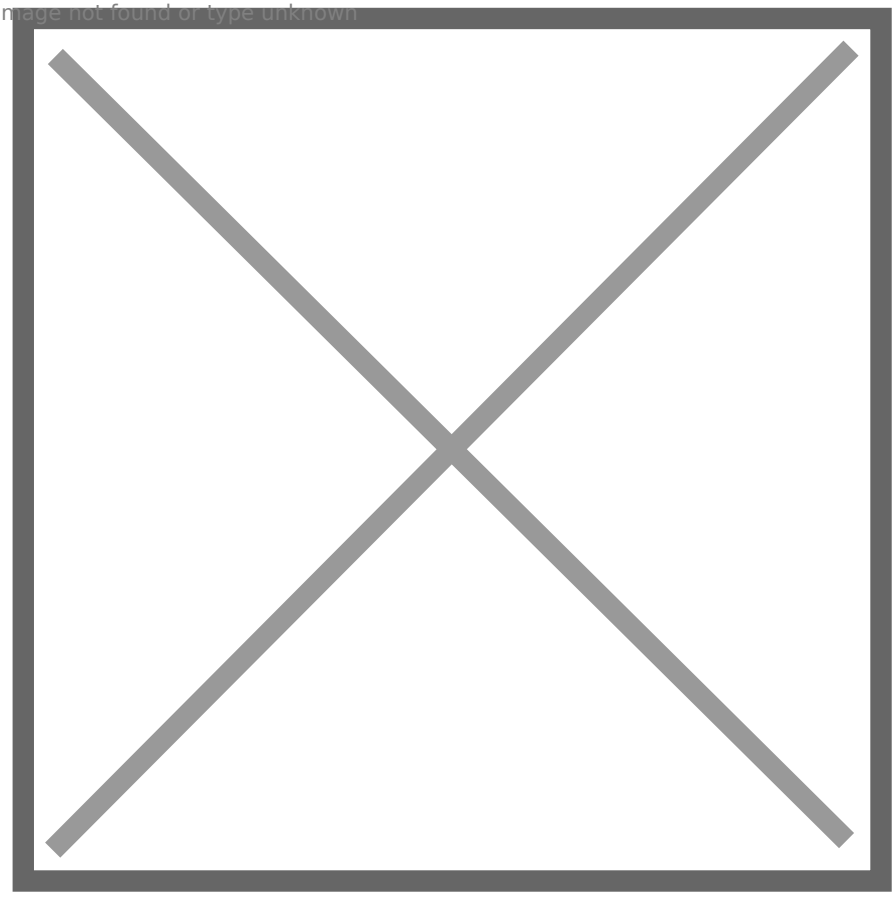
By this way, we can abuse impersonate right to get sysadmin privilege. Let’s check if we configured correctly. First, if we successfully integrate Kerberos authentication. Import powerupsql.ps1 script, and enumerate domain instance.

Image not found or type unknown



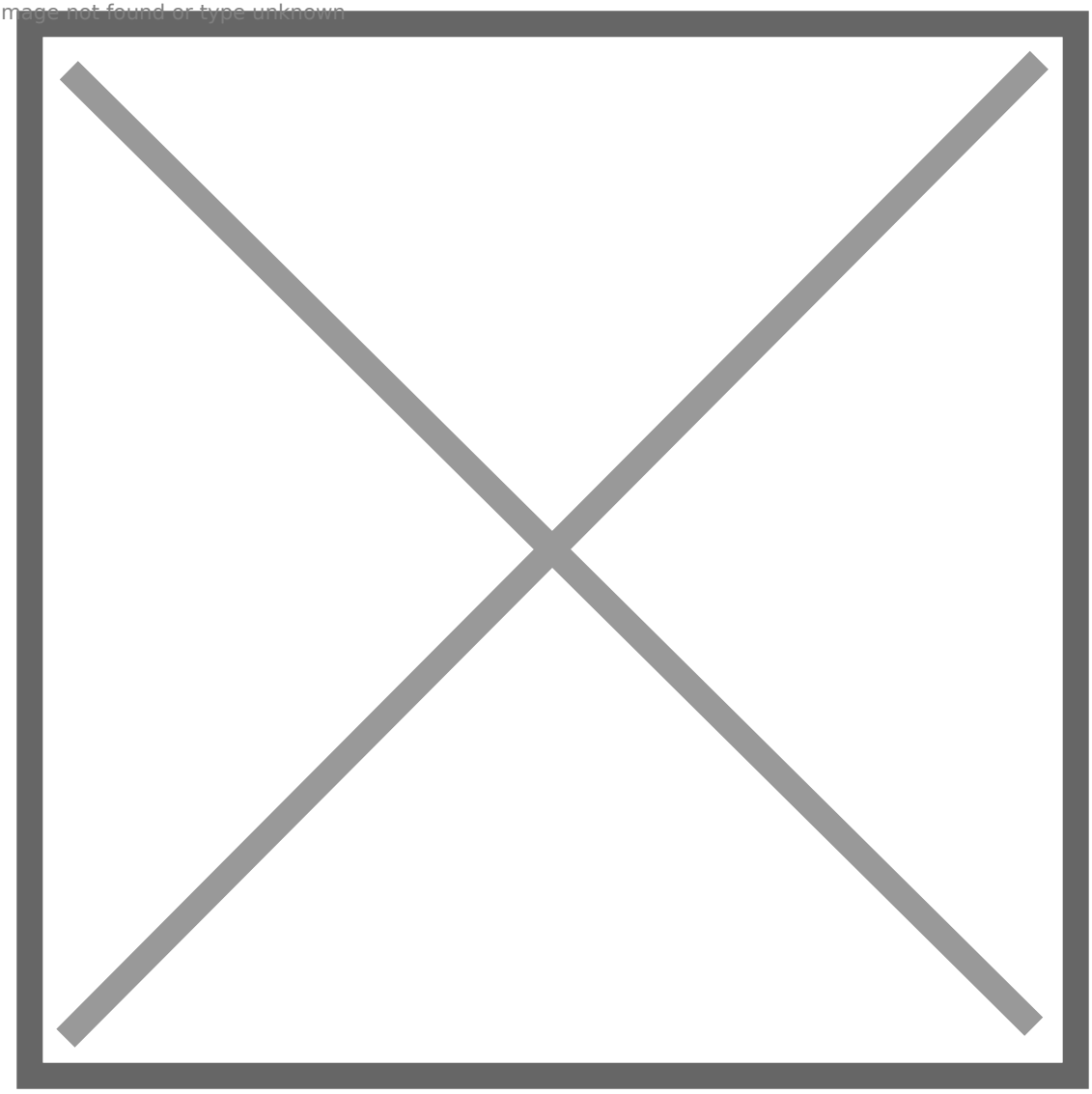
It looks great! Then, access any instance to check if we get a TGS for SQL service. Here, I tested srv01.

Image not found or type unknown



Check cached tickets, and I find the TGS, it means Kerberos is integrated successfully.

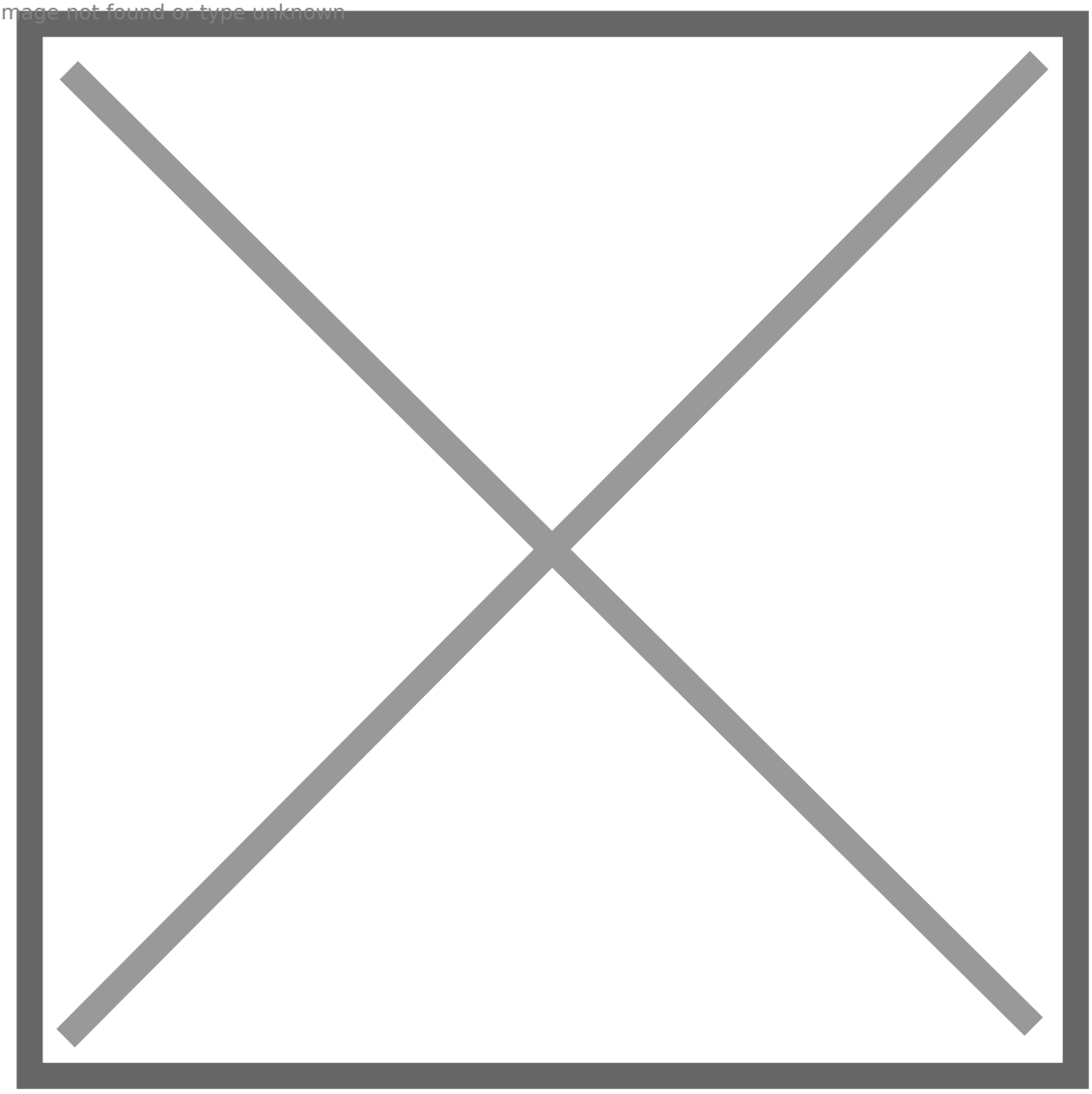
Image not found or type unknown



Then, we need to verify permission assignment. I choose three types of users to check

helen.park: Least privilege

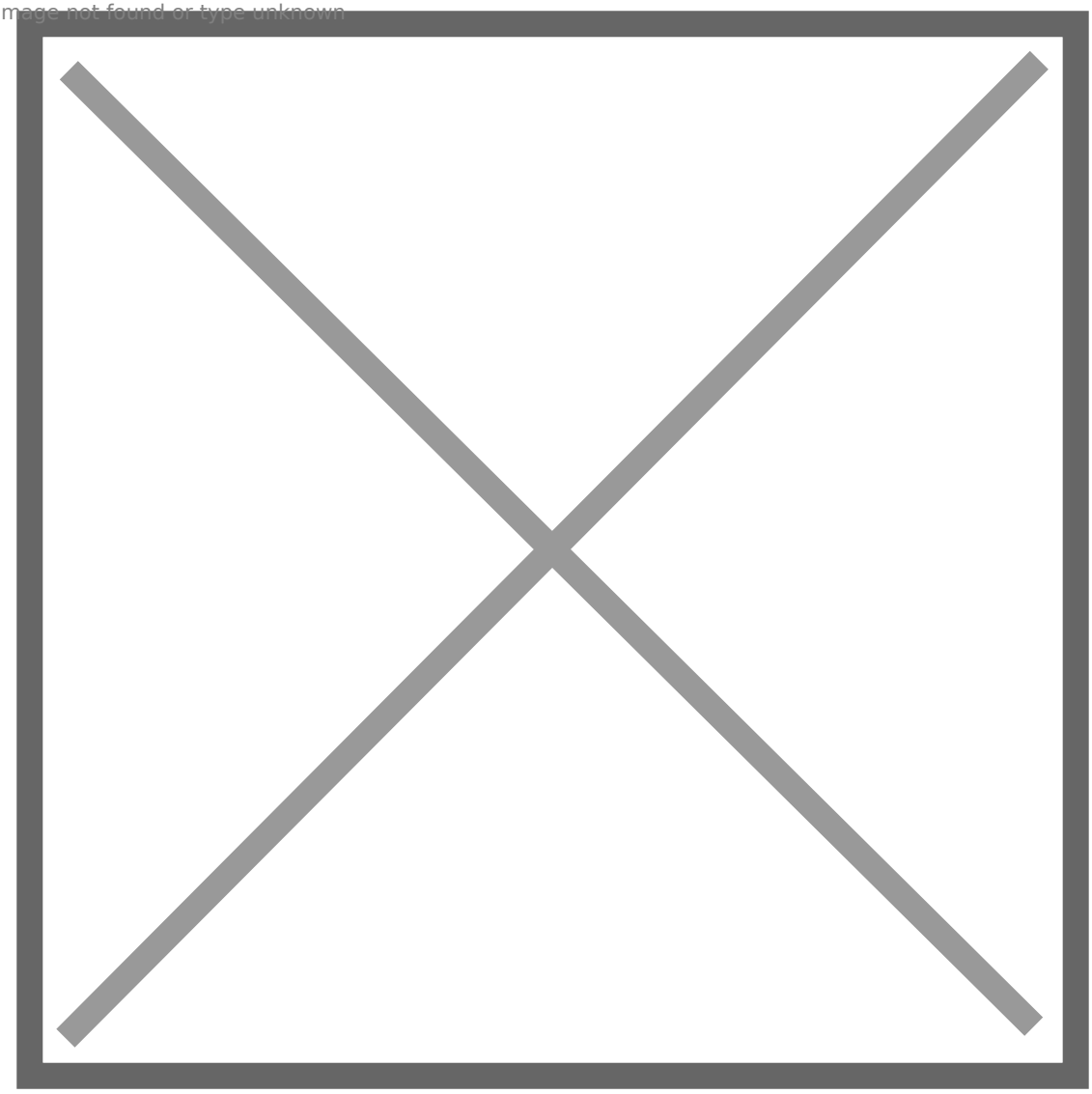
Image not found or type unknown



We can see, helen.park can only access SQL instance and has very limited privilege. She cannot impersonate other logins.

Administrator: Sysadmin

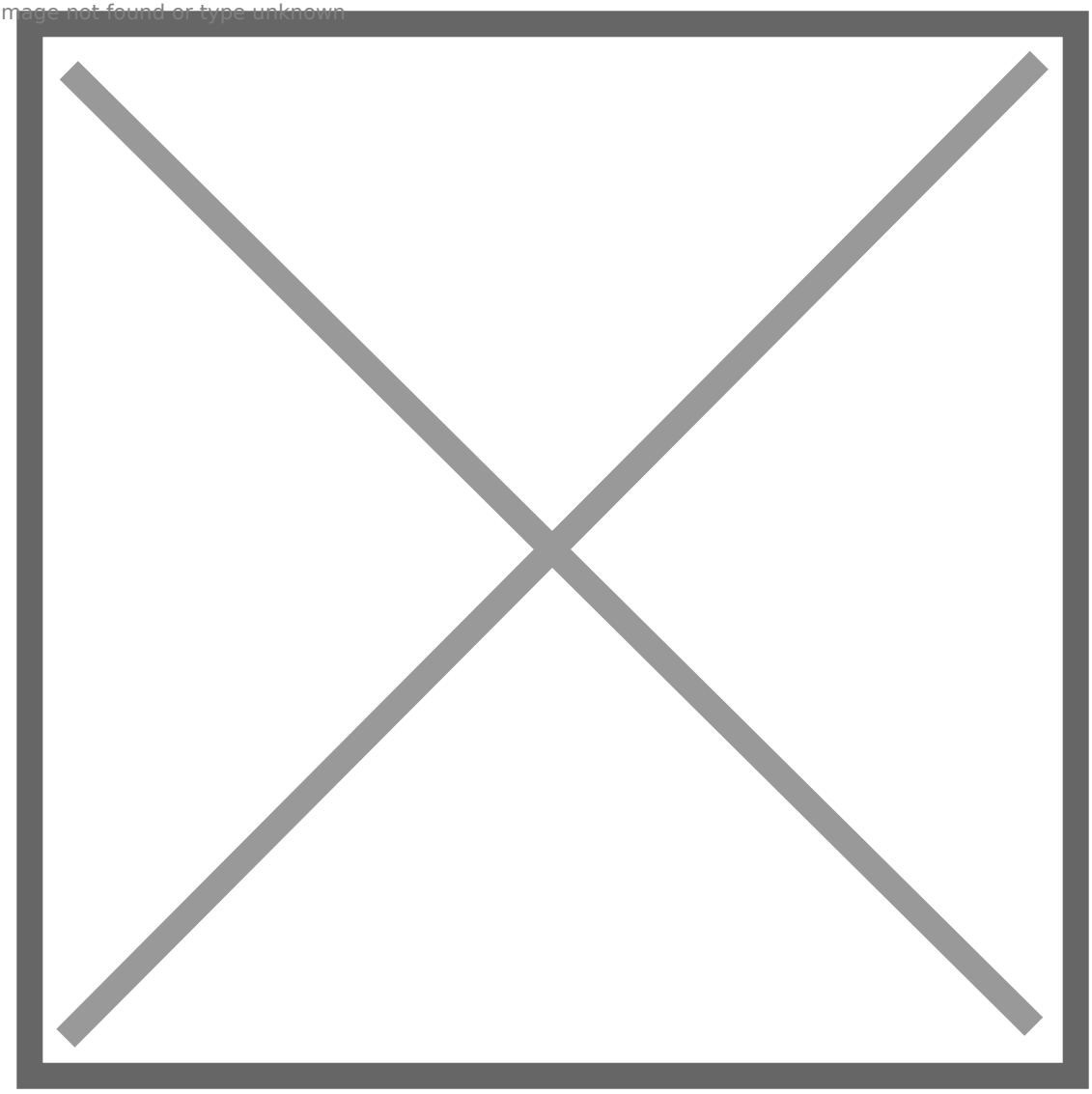
Image not found or type unknown



Domain admin has highest privilege.

svc_sql: Can impersonate sa to get sysadmin privilege.

Image not found or type unknown



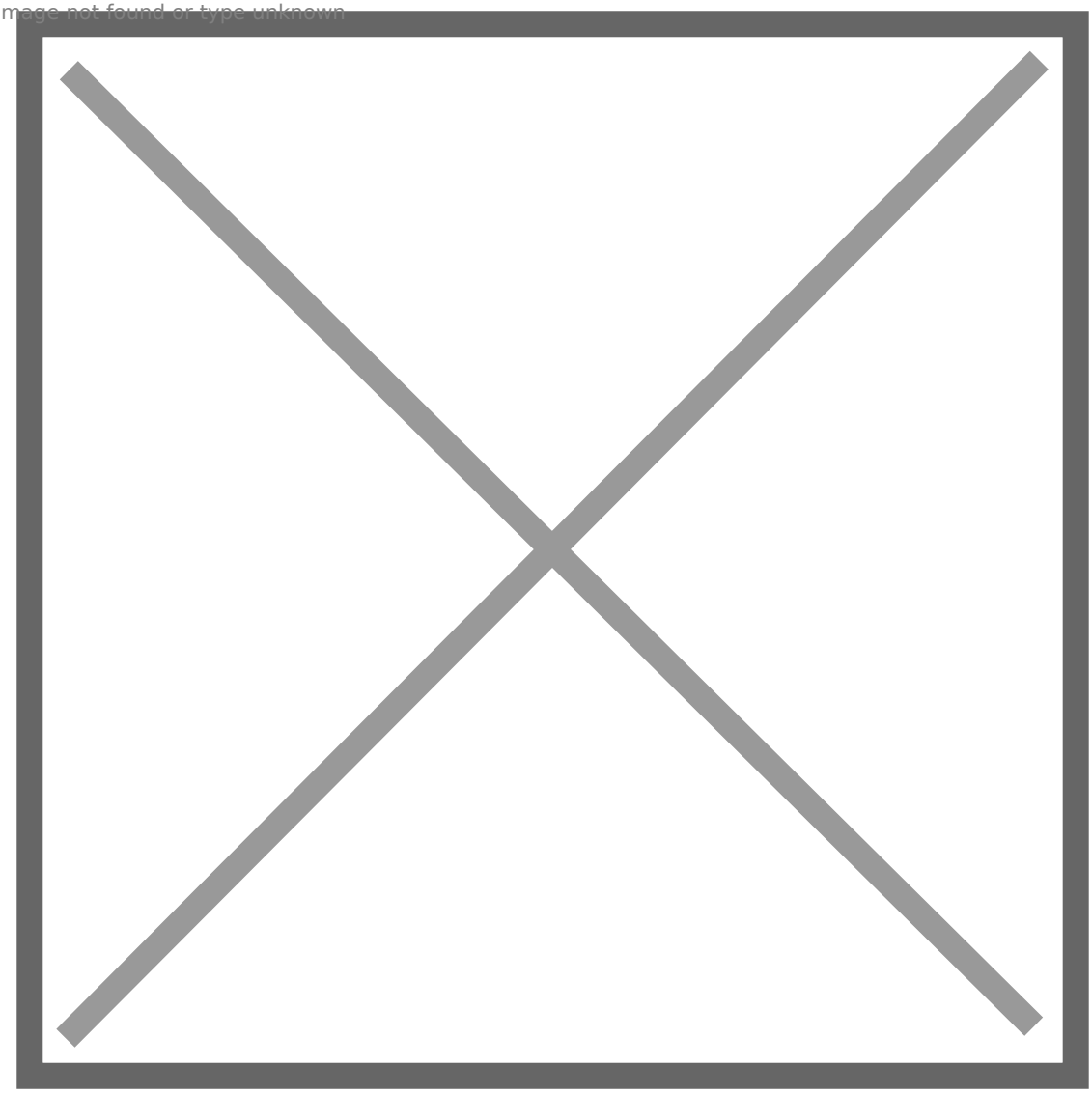
After impersonation, svc_sql does not have sysadmin privilege. But after impersonate, it has sysadmin privilege.

So the permission assignment is successful as well.

Up to now, we can repeat previous steps related in SQL part on SRV02, but just remember to change server/instance value. But then we will configure SQL link on SRV01, I did not configure SQL link on SRV02, but of course you can add one.

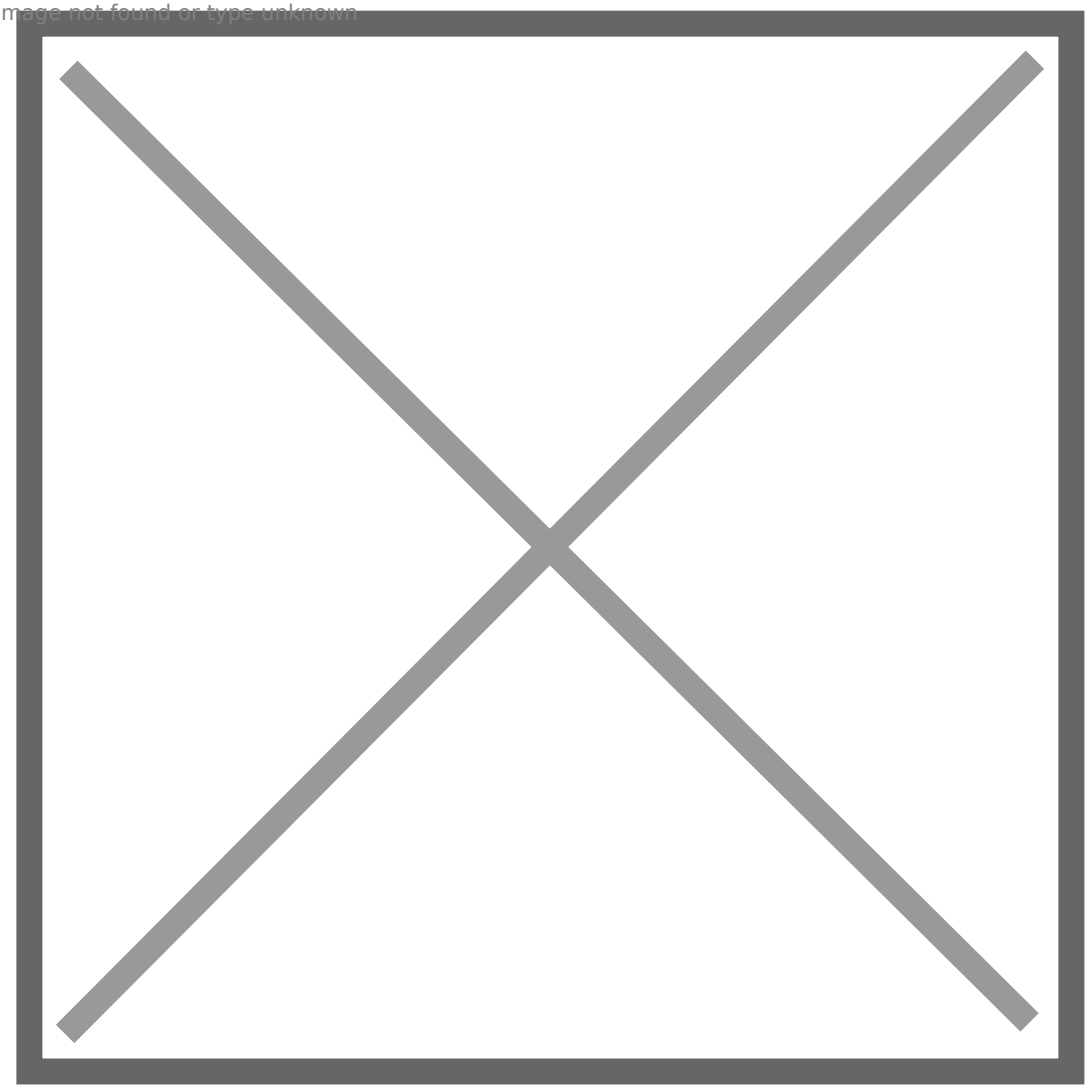
Right click **Server Objects -> Linked Servers,** add a new link. The **General** tab should be like this:

Image not found or type unknown



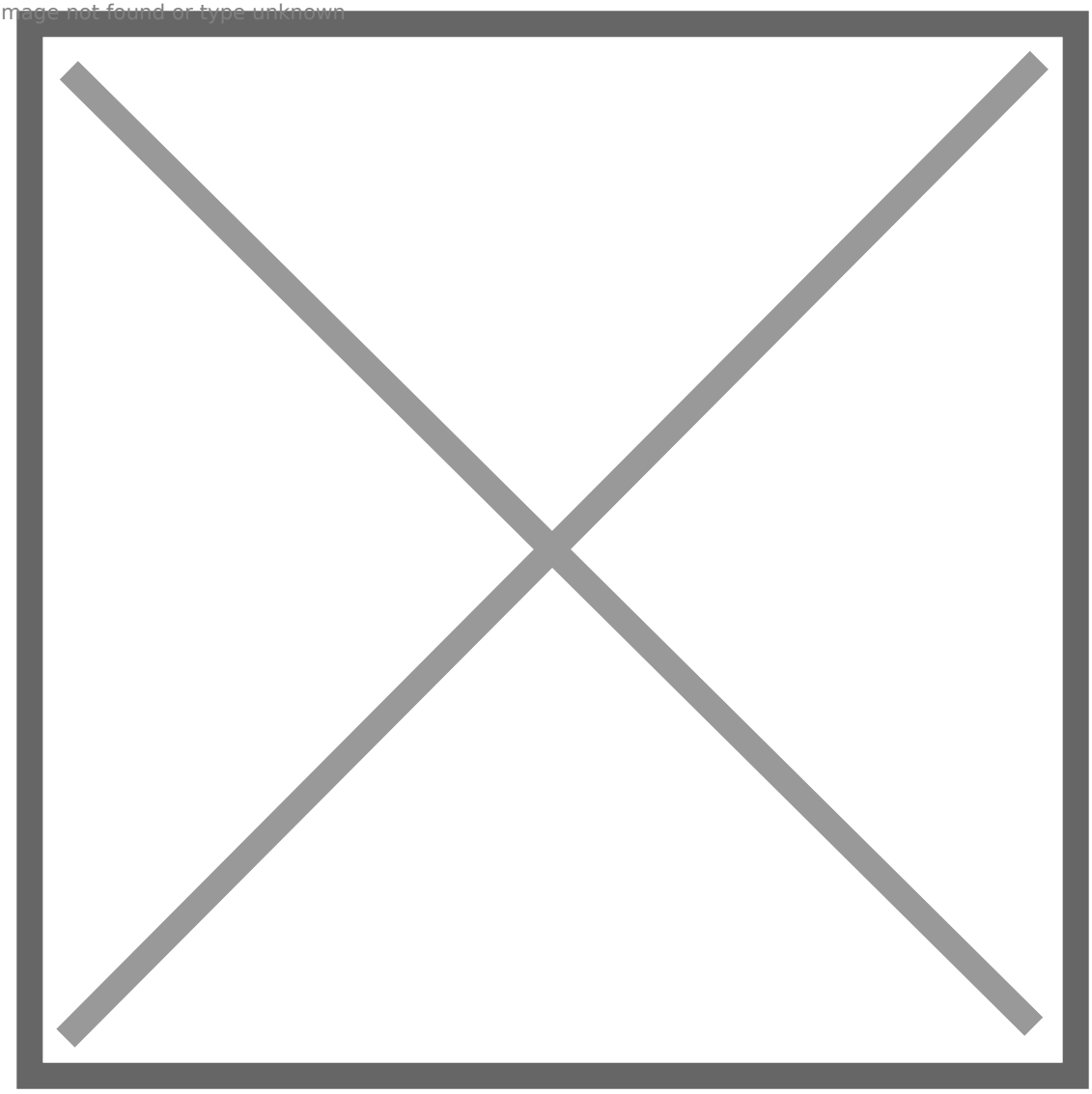
On **Security** tab, we add a new entry to login mappings, map local login sa to remote login sa. If it is confusing, you can change map to SRV02\Administrator. What does it mean? If our current login is sa, we know we have sysadmin privilege on SRV01. But if we follow the link to reach SRV02, we may not have sysadmin privilege. Since we are designing a misconfiguration, so I just map it to an sysadmin login on SRV02. By this way, we still have sysadmin login when reaching SRV02. And select “Be made using the login’s current security context” option, it is easy to understand.

Image not found or type unknown



Okay, so we configured SQL link: SRV01 -> SRV02, let’s check it.

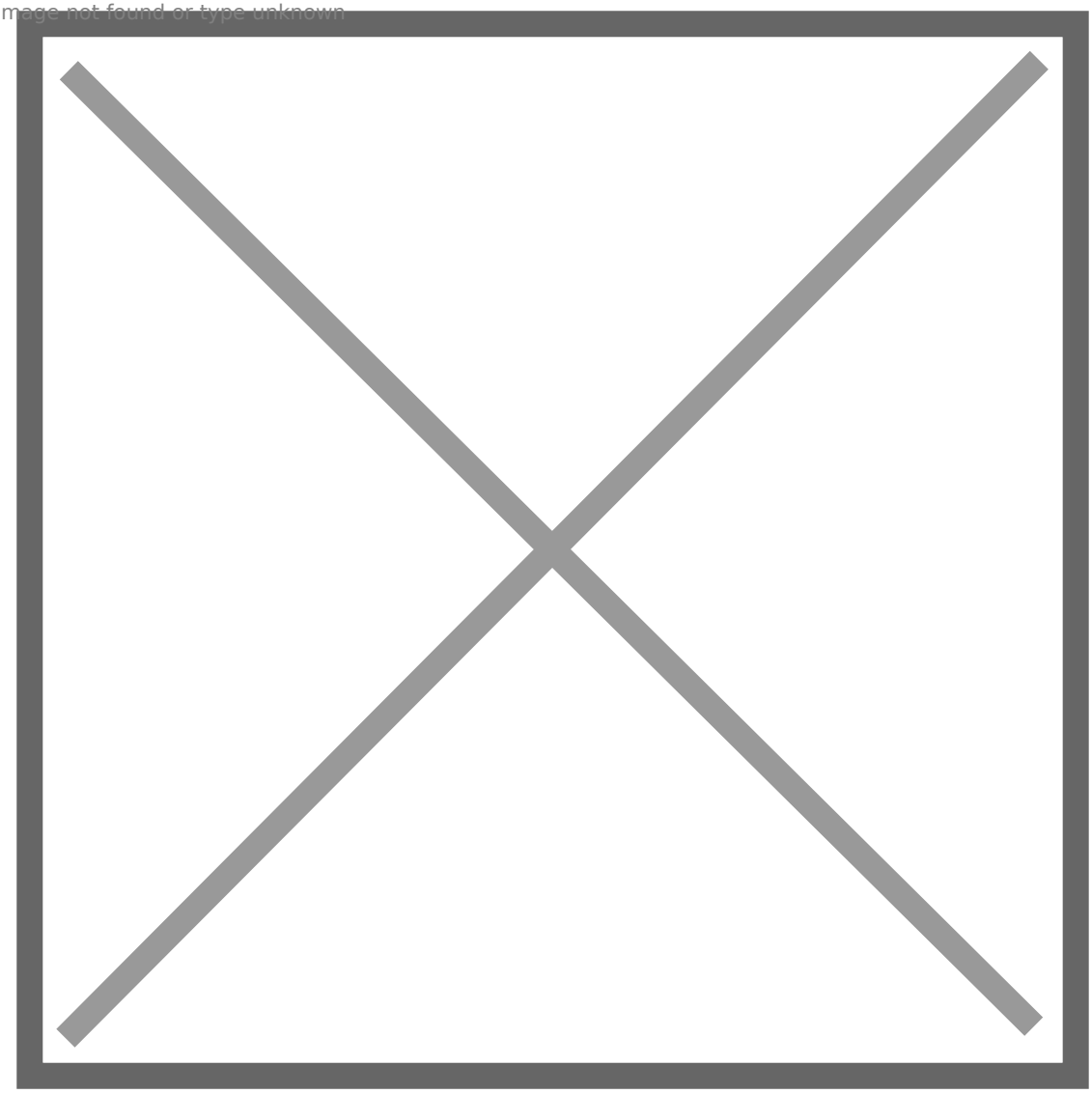
Image not found or type unknown



The link is correct, then let’s check if we can still have sysadmin privilege on SRV02 via SQL link.

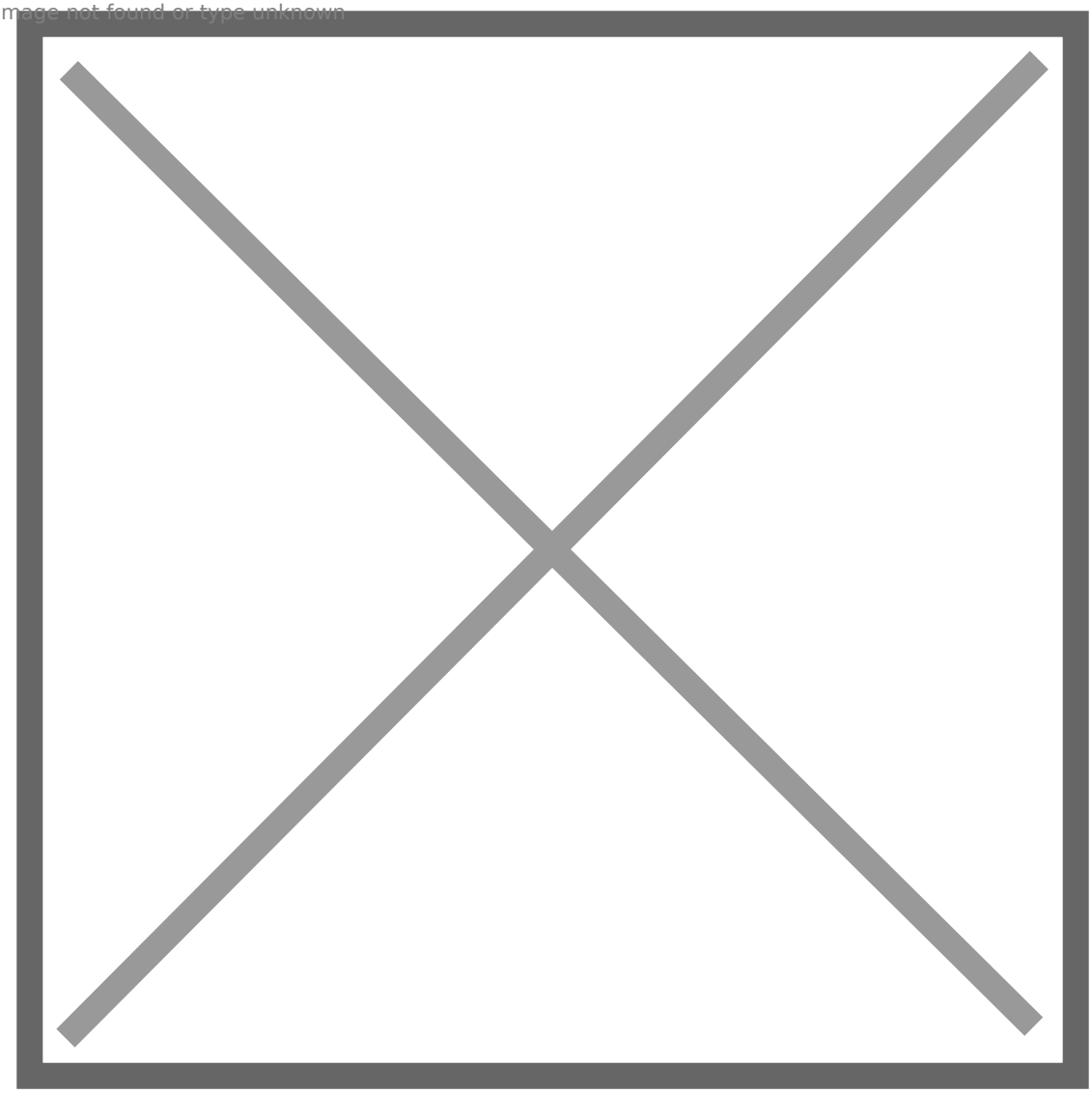
Before impersonation:

Image not found or type unknown



We can see if we do not impersonate sa and follow the link, we will get an error, because we did not map svc_sql to SRV02 previously. However, if we impersonate sa, then the result it totally different.

Image not found or type unknown



We can access SRV02 with sysadmin privilege! So the permission is configured well.

After a long journey, we successfully configured SRV01.

SERVER 2

srv02.blackops.local

Autologin: None

Remote Desktop Login: Enable RDP

SQL Instance: Almost the same as we did on SRV01, but with a different IP/Instance. And no need to add a link, but if you want, that’s totally cool as well.

We previously have set unconstrained delegation for SRV02.

Enable **PPL** for SRV02 as well.

We finally successfully built the whole vulnerable AD set!

IN THE END

Thanks for spending time on reading such a long article, I really appreciate! This is the first time for me to design a vulnerable AD set, so there is a lot of room for improvement. And though the guide is very detailed, I cannot make sure I did not miss anything. If you follow my steps and still have difficulty making it work, just let me know!

In the future, if I plan to design more vulneable AD sets, I would like to cover and add more features and vectors such as ADCS abuse, Relay Attack, Phishing and User Simulation, etc.

Since there is copyright concern, I will make sure if it is legal to share my VM/images. If it is okay, I will share my own VM soon. But building by your own is a good way to learn! I will release the walkthrough of the vulnerable AD soon. I invited my friend (Passed OSCP, CRTO) to test my vulnerable AD set, he reached the 3rd machine after about 24 hours with some hints. And after about 72 hours, he reached DC. Welcome to play with my AD set : D

Update: Walkthrough of this AD set: https://gustavshen.medium.com/walkthrough-of-my-vulnerable-ad-set-d56abeae5bac

Few bug fixes/updates have been made into the article.

If you think my article is helpful for you, buying me a coffee is always appeciated (**ko-fi.com/senzee**)**!**

Revision #1
Created 28 February 2024 18:20:55 by winslow
Updated 28 February 2024 19:38:59 by winslow