

Armand OCTEAU – Student 2422790

Finished on Saturday 19 April 2025

NETWORK OPERATING SYSTEMS AND SERVICES (NOSS)

- Research, build, and deploy, a Virtualized network –

Summary

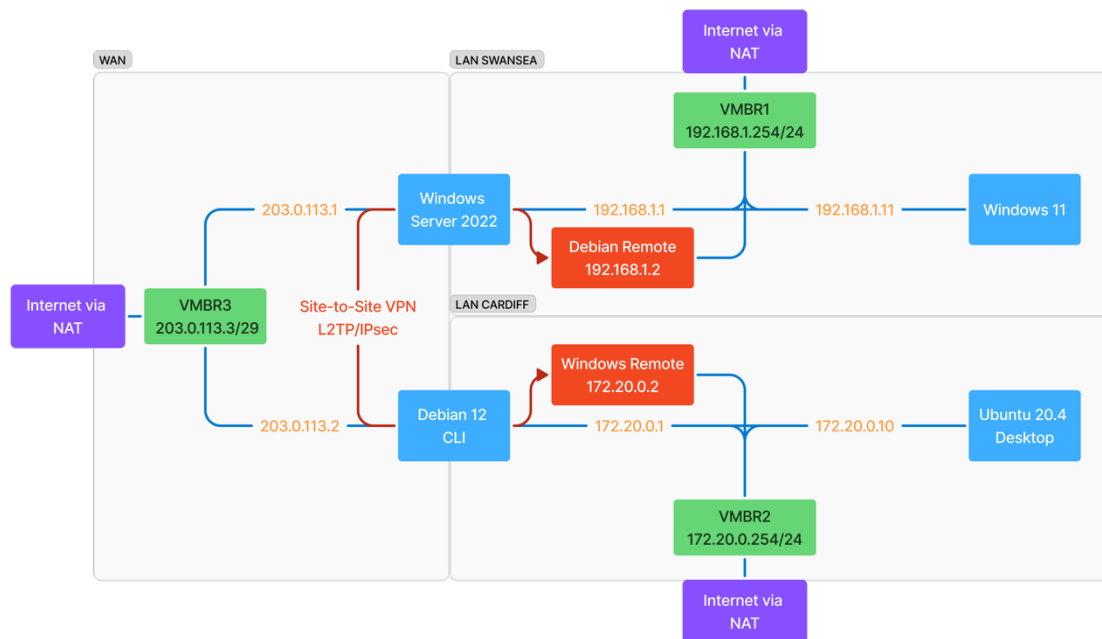
TASK 1	1
Site-to-Site VPN Configuration.....	2
- <i>New Topology</i> -	2
- <i>Configuration on Windows Server</i> –	3
- <i>Configuration on Debian</i> -	4
- <i>Static Routes configuration</i> -	5
Active Directory Configuration for Swansea	7
- <i>Forest creation</i> -	7
- <i>DNS Server Configuration</i> -	8
- <i>Testing on Swansea's LAN</i> -	10
Active Directory Configuration for Cardiff	12
- <i>Child domain creation</i> –	12
- <i>Domain Trust</i> –	13
- <i>Testing on Cardiff's LAN</i> –	14
Importing AD Users	16
- <i>On Swansea's DC</i> -	16
- <i>On Cardiff's DC</i> -	16
Shared Folder on Swansea's	19
- <i>Creating the new disk</i> -	19
- <i>Setting permissions</i> -	20
- <i>Testing</i> -	21
Configuring IIS's server	21
- <i>Configuring</i> –	21
- <i>Testing</i> –	23
- <i>Issue Resolving</i> –	24
Remote Desktop Services	25
- <i>Configuring</i> –	25
- <i>Issue Resolving</i> –	26
- <i>Creating the collection</i> –	26
- <i>Testing</i> –	27

TASK 1

- Further configuration -

Site-to-Site VPN Configuration

- *New Topology* -



1 - New topology of the network

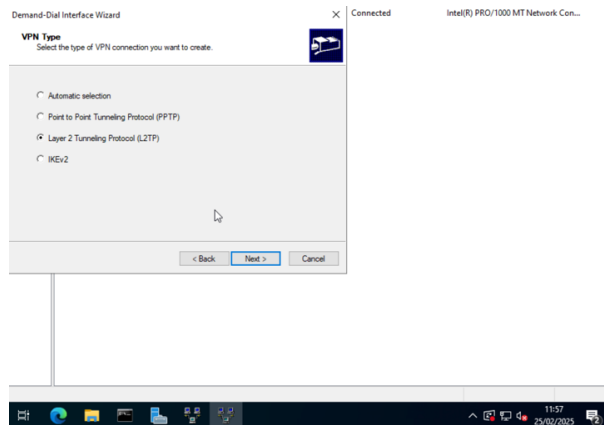
First, let's have a look at the new topology of the network. For this coursework, we'll be keeping the configuration last made during Coursework 1 as a starting point.

As we can see in the diagram above, a site-to-site VPN will be configured between the two LAN servers, allowing each LAN to access the other over a secure tunnel. Debian Remote is the Debian server on Swansea's LAN, and conversely, Windows Remote is the Windows Server on Cardiff's LAN.

During Coursework 1, ICMPv4 traffic was already allowed in the advanced firewall settings to enable ping and complete the coursework. The DHCP Server was also already configured and installed on both servers. Therefore, only the Routing and Remote Access Service (RRAS) was then installed on the WS22 VM, with VPN being selected.

To configure the site-to-site VPN, one had to decide which server would act as the initiator. In this configuration, Windows Server is the main server that hosts the shared folder, main Active Directory domain, web server, and RDS. Therefore, it was the best choice to act as the VPN tunnel initiator.

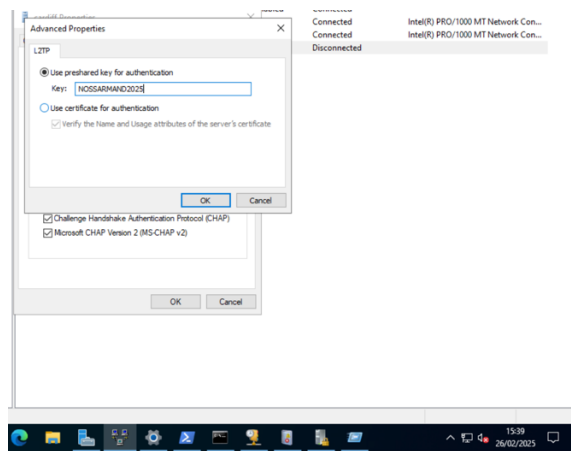
- Configuration on Windows Server –



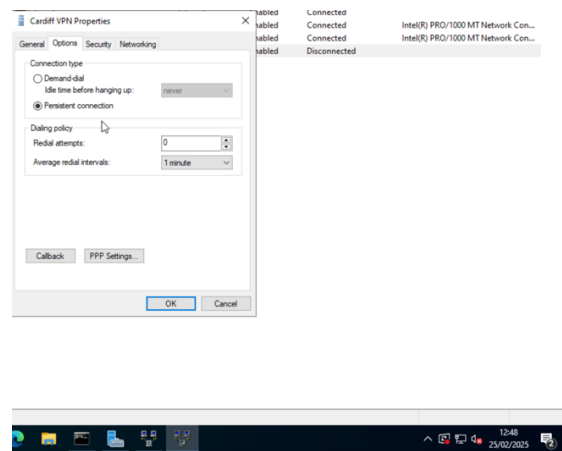
2 - RRAS Configuration step 1

In order to configure the site-to-site VPN, L2TP/IPsec was chosen as the protocol, as it is a secure and widely used option. IPsec authentication was configured to use a shared PSK key (chosen to be NOSSARMAND2025) instead of a certificate-based authentication. The L2TP tunnel will be authenticated with credentials to the Debian server using the ppp package.

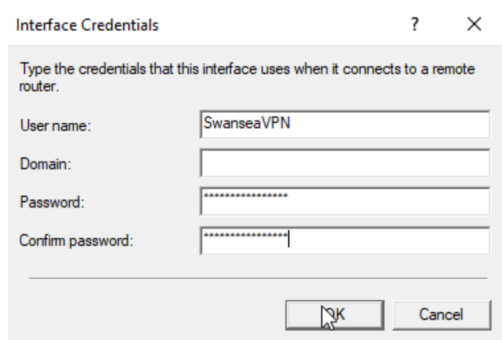
A new Demand-Dial interface called “Cardiff” was then created with the following configuration, allowing Windows Server to initiate the connection to Debian.



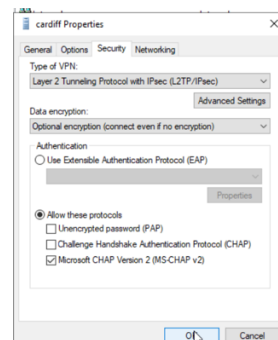
3 - VPN configuration : setting the shared PSK key



4 - VPN configuration : setting to persistent connection



5 - VPN configuration : IPsec credentials for Debian



6 - VPN configuration : settings card of the interface

- Configuration on Debian -

Now, let's configure Debian so that it can allow Windows Server to connect. The following packages were installed: strongswan-ipsec, xl2tpd, and ppp. IPsec is responsible for securing the traffic using the shared PSK key over the IKE protocol, xl2tpd handles the tunnel traffic, and ppp manages remote authentication.

```
conn Swan-Card
    auto=add
    keyexchange=ikev1
    authby=secret
    ike=aes128-sha1-modp1024
    esp=aes128-sha1

    left=203.0.113.2
    leftsubnet=172.20.0.0/24
    leftid=203.0.113.2
    leftprotoport=17/1701

    right=203.0.113.1
    rightsubnet=192.168.1.0/24
    rightid=203.0.113.1
    rightprotoport=17/1701

    type=tunnel
    dpdaction=clear
    dpddelay=30
    dpdtimeout=120

root@debianserver-ca:~#
root@debianserver-ca:~# date
Fri 7 Mar 20:36:58 GMT 2025
```

7 - VPN configuration: ipsec.conf

```
[global]
listen-addr = 203.0.113.2

[lns default]
local ip = 172.20.0.1
require authentication = yes
name = cardiff
assign ip = yes
ip range = 172.20.0.2-172.20.0.3
refuse pap = yes
require chap = yes
ppp debug = yes
pppoptfile = /etc/ppp/options.xl2tpd
length bit = yes

root@debianserver-ca:~# date
Fri 7 Mar 20:38:05 GMT 2025
root@debianserver-ca:~# _
```

8 - VPN configuration: xl2tpd.conf

```
require-mschap-v2
ipcp-accept-local
ipcp-accept-remote
ms-dns 1.1.1.1
ms-dns 8.8.8.8
auth
mtu 1400
mru 1400
ipcp-accept-local
ipcp-accept-remote
debug
nocc
refuse-pap
refuse-chap
logfile /var/log/ppp.log
connect-delay 5000
nobsdcomp

root@debianserver-ca:~# date
Fri 7 Mar 20:38:55 GMT 2025
```

9 - VPN configuration:
ppp/options.xl2tpd

The credentials were also set in ipsec.secrets for the PSK key and in /ppp/chap-secrets for the username and password. IPsec is configured to use the IKEv1 key exchange protocol; otherwise, Windows Server would not accept the PSK key. The encryption was also set to match the one used on Windows Server.

For both servers, an IP pool of two IPs was set: 0.0.0.2-3 for each subnet, ensuring that the connected desktop receives the second IP address of the LAN's subnet (.2), as shown in the previously displayed diagram.

Now, the VPN can connect on Windows Server's side but does not work on Debian's side. Effectively, the same IP being 172.20.0.1 was assigned to both the ppp0 (the vpn) and the ens19 (the lan) interfaces, resulting in confusion for the server. To resolve this, ARP proxy was enabled in sysctl.conf. This task is done automatically on Windows Server, but Linux, being a more low-level operating system, does not handle it by default.

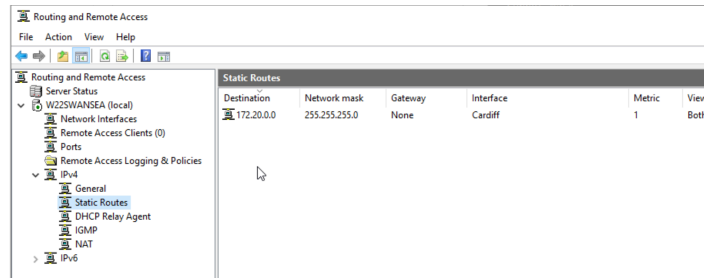
For Swansea's LAN and Cardiff's LAN clients to communicate with each other, static routes had to be set on both sides. On Windows Server, a static route was easily added through the GUI to direct all traffic bound for 172.20.0.0/24 through the VPN interface. However, on Debian, a bash script had to be written in the /etc/ppp/ip-up.d/ folder. This ensures that when the VPN tunnel is initiated, the route is automatically created—and deleted when the tunnel is brought down.

- Static Routes configuration -

```
#!/bin/sh -e
exec ip route add 192.168.1.0/24 dev $PPP_IFACE
exec ip route add 172.20.0.2 dev $PPP_IFACE
exit 0

root@debianserver-ca:~# date
Fri 7 Mar 21:03:47 GMT 2025
```

10 - Debian Script to automatically add the static route to Swansea

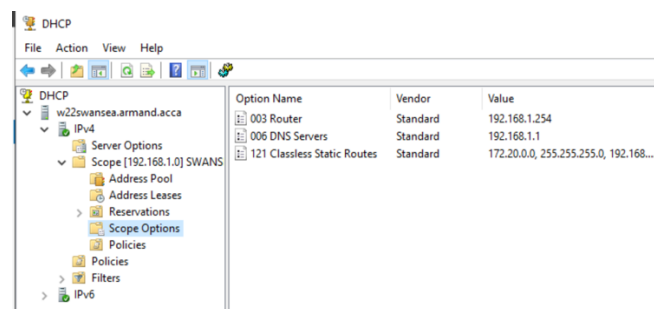


11 - Windows static route to cardiff configuration

In our topology, Proxmox's Linux bridges act as routers and internet providers via NAT rules for each LAN. Therefore, on each client, the gateway is defined as the Linux bridge, not the LAN server. To achieve this and to provide clients the static route to the remote LAN, a 121 classless-static-routes rule had to be researched and configured in the DHCP server, to be sure that requests from Swansea's LAN to 172.20.0.0/24 pass through 192.168.1.1 instead of 192.168.1.254—the same applies to Cardiff's LAN with its corresponding IPs.

On Windows Server, this rule was easily added through the DHCP Manager, as shown in the following screenshots. However, on Debian, the process was more difficult as expected. The rule did not work with the isc-dhcp-server package, which had been previously installed and configured. This issue went up because the package has been discontinued since 2022.

Therefore, the kea-dhcp4-server package was installed as a replacement. This is the latest DHCP server package developed and maintained by ISC for Linux. The entire DHCP server was reconfigured just as before, with the 121 static route rule successfully added.



12 - 121 Classless Static Routes rule on WS

```

"Dhcp4": {
  "interfaces-config": {
    "interfaces": ["ens19"]
  },
  "lease-database": {
    "type": "memfile"
  },
  "subnet4": [
    {
      "subnet": "172.20.0.0/24",
      "pools": [
        { "pool": "172.20.0.10 - 172.20.0.100" }
      ],
      "option-data": [
        {
          "name": "routers",
          "data": "172.20.0.254"
        },
        {
          "name": "domain-name-servers",
          "data": "172.20.0.2"
        },
        {
          "name": "broadcast-address",
          "data": "172.20.0.255"
        },
        {
          "name": "classless-static-routes",
          "space": "dhcp4",
          "code": 121,
          "data": "24,192,168,1,172,20,0,1"
        }
      ]
    }
  ],
  "option-def": [
    {
      "name": "classless-static-routes",
      "code": 121,
      "space": "dhcp4",
      "type": "record",
      "record-types": "uint8,uint8,uint8,uint8,uint8,uint8,uint8,uint8"
    }
  ]
}
}

root@DEBIANSERVER-CARDIFF:~# date
Sun  2 Mar 19:31:59 GMT 2025
root@DEBIANSERVER-CARDIFF:~#

```

13 - New DHCP config with Kea with the 121 static-route rule

Now, the site-to-site VPN is fully operational and working. Each clients and servers can ping each others, and the traffic is routing as it should.

```

Microsoft Windows [Version 10.0.20348.3207]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Administrator>ipconfig

Windows IP Configuration

Ethernet adapter WAN:

    Connection-specific DNS Suffix  . : 
    Link-local IPv6 Address . . . . . : fe80::e8d2:83ca:7871:b67e
    IPv4 Address. . . . . : 203.0.113.1
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 203.0.113.3

Ethernet adapter LAN:

    Connection-specific DNS Suffix  . : 
    Link-local IPv6 Address . . . . . : fe80::d22a:4aa:f1cb:5fd5
    IPv4 Address. . . . . : 192.168.1.1
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 

PPP adapter Cardiff:

    Connection-specific DNS Suffix  . : 
    IPv6 Address. . . . . : ::1c82:467:fd96:1b8a
    Link-local IPv6 Address . . . . . : fe80::1c82:467:fd96:1b8a
    IPv4 Address. . . . . : 172.20.0.2
    Subnet Mask . . . . . : 255.255.255.255
    Default Gateway . . . . . : 

PPP adapter RAS (Dial In) Interface:

    Connection-specific DNS Suffix  . : 
    IPv6 Address. . . . . : ::1bc83:1a78:3361:b639
    Link-local IPv6 Address . . . . . : fe80::1bc83:1a78:3361:b639
    IPv4 Address. . . . . : 192.168.1.2
    Subnet Mask . . . . . : 255.255.255.255
    Default Gateway . . . . . : 

C:\Users\Administrator>

```

```

root@debianserver-ca:~# ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host noprefixroute
        valid_lft forever preferred_lft forever
2: ens18: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether bc:24:11:ec:ab:5d brd ff:ff:ff:ff:ff:ff
    altname enp0s18
    inet 203.0.113.2/29 brd 203.0.113.7 scope global ens18
        valid_lft forever preferred_lft forever
    inet6 fe80::be24:11ff:feec:ab5d/64 scope link
        valid_lft forever preferred_lft forever
3: ens19: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether bc:24:11:00:fd:0f brd ff:ff:ff:ff:ff:ff
    altname enp0s19
    inet 172.20.0.1/24 brd 172.20.0.255 scope global ens19
        valid_lft forever preferred_lft forever
    inet6 fe80::be24:11ff:fe00:fd0f/64 scope link
        valid_lft forever preferred_lft forever
4: ppp0: <POINTOPOINT,MULTICAST,NOARP,UP,LOWER_UP> mtu 1400 qdisc fq_codel state UNKNOWN group default
    link/ppp
    inet 172.20.0.1 peer 172.20.0.2/32 scope global ppp0
        valid_lft forever preferred_lft forever
    inet6 ::fdfb:2c87:ef6d:209e/64 scope global dynamic mngtmpaddr
        valid_lft 2591768sec preferred_lft 604568sec
    inet6 fe80::fdfb:2c87:ef6d:209e peer fe80::1c82:467:fd96:1b8a/128 scope link
        valid_lft forever preferred_lft forever

root@debianserver-ca:~# date
Fri  7 Mar 20:59:02 GMT 2025
root@debianserver-ca:~#

```

15 - VPN configuration: successfully connected on Debian

14 - VPN configuration: successfully connected on WS

The screenshot shows two windows side-by-side. The left window is a Command Prompt titled 'Command Prompt' with the following text:

```
Microsoft Windows [Version 10.0.26100.3476]
(c) Microsoft Corporation. All rights reserved.

C:\Users\A.octeau>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:
Reply from 192.168.1.1: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.1:
    Packets: Sent = 1, Received = 1, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
Control-C
^C

C:\Users\A.octeau>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:
Reply from 192.168.1.2: bytes=32 time<1ms TTL=127

Ping statistics for 192.168.1.2:
    Packets: Sent = 1, Received = 1, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
Control-C
^C

C:\Users\A.octeau>ping 172.20.0.1

Pinging 172.20.0.1 with 32 bytes of data:
Reply from 172.20.0.1: bytes=32 time<1ms TTL=63

Ping statistics for 172.20.0.1:
    Packets: Sent = 1, Received = 1, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 0ms, Average = 0ms
Control-C
^C
```

The right window is a PowerShell window titled 'Windows PowerShell' with the following text:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! ht
tps://aka.ms/PSWindows

PS C:\Users\A.octeau> ping 172.20.0.10

Pinging 172.20.0.10 with 32 bytes of data:
Reply from 172.20.0.10: bytes=32 time=1ms TTL=62

Ping statistics for 172.20.0.10:
    Packets: Sent = 1, Received = 1, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 1ms, Average = 1ms
Control-C

PS C:\Users\A.octeau> ping 203.0.113.1

Pinging 203.0.113.1 with 32 bytes of data:
Request timed out.

Ping statistics for 203.0.113.1:
    Packets: Sent = 1, Received = 0, Lost = 1 (100% loss),
Control-C

PS C:\Users\A.octeau> ping 203.0.113.2

Pinging 203.0.113.2 with 32 bytes of data:
Request timed out.

Ping statistics for 203.0.113.2:
    Packets: Sent = 1, Received = 0, Lost = 1 (100% loss),
Control-C
PS C:\Users\A.octeau> |
```

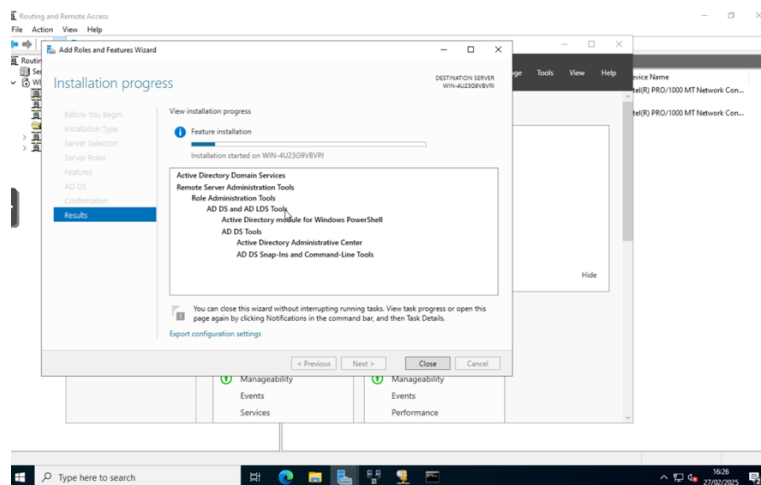
At the bottom of the PowerShell window, there is a message: 'Activate Windows Go to Settings to activate Windows.'

We can see that from the Windows 11 client we can ping Windows Server, Debian Remote, Debian, Ubuntu, but not Debian and Windows Server on the WAN. As it should.

Active Directory Configuration for Swansea

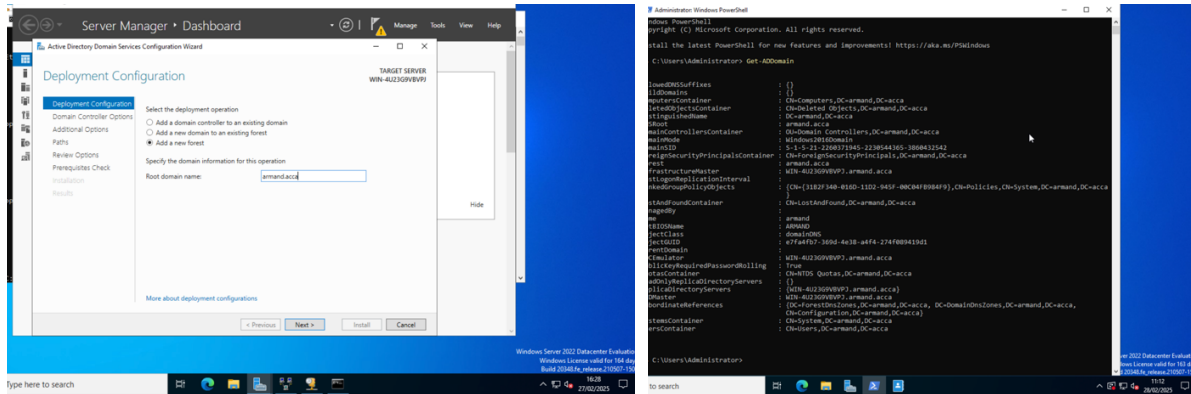
- Forest creation -

First, we'll be configuring the parent domain on Swansea's side to ensure it works perfectly before adding the child domain in Cardiff's.



16 - Active Directory role installation on WS

Once the roles were installed, Windows Server was promoted as a Domain Controller (DC) and the forest **armand.acca** was then created as shown below.

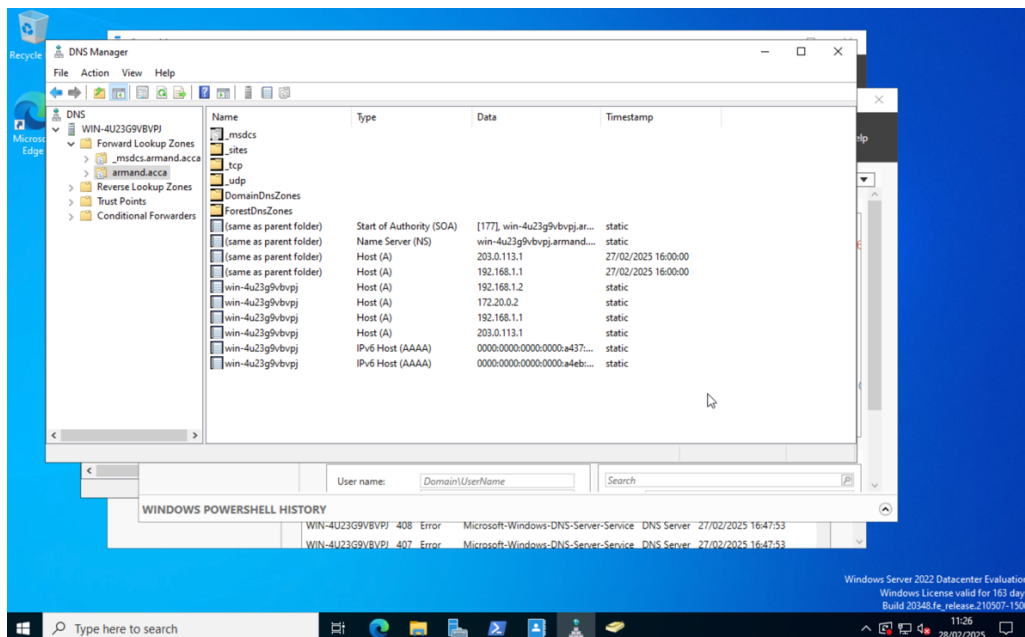


17 - armand.acca forest creation

18 - Parent domain successfully created

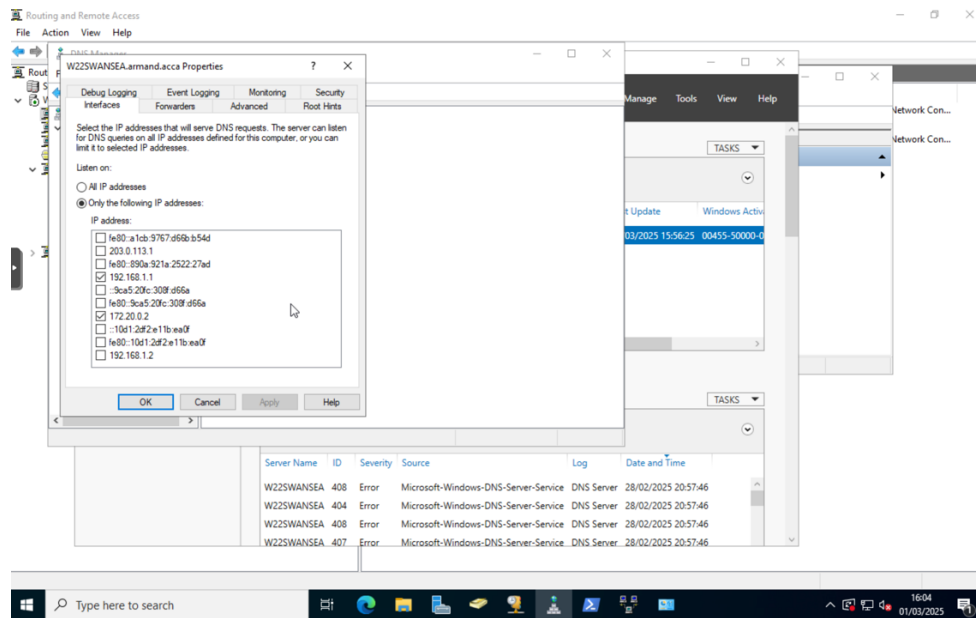
- DNS Server Configuration -

Let's configure the DNS Server. As it's possible to see, a lot of unwanted IP's zones were created such as the Windows Server WAN's IP and it's IPv6 IPs (which we don't use in this Coursework).



19 - Unwanted Hosts in the DNS Server on WS

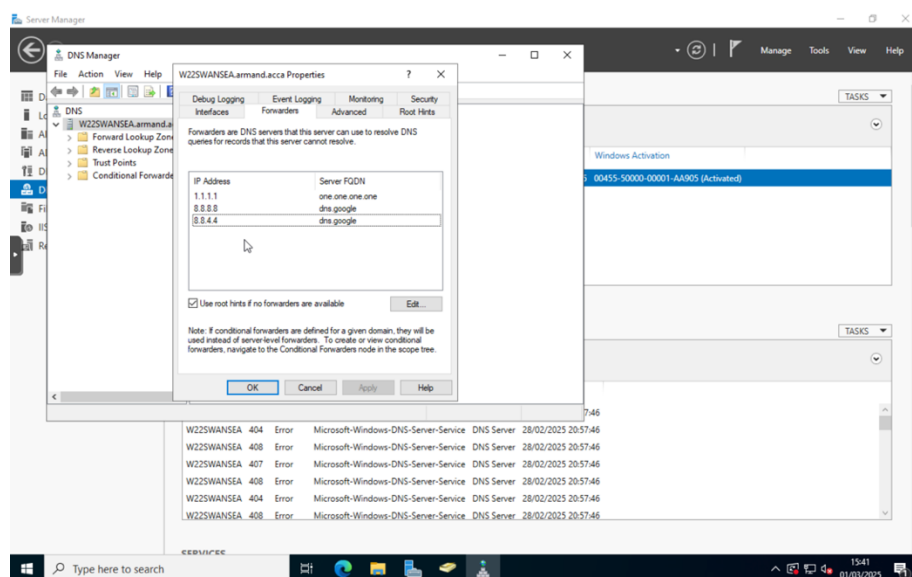
To solve this issue, the DNS Server was assigned to listen only on the LAN's interface and on the remote VPN interface in its properties, as we don't want the forest to be accessible on the WAN. A reverse PTR lookup zone was also created for 192.168.1.0 (as shown in a later screenshot).



20 - Interfaces on which the DNS Server is listening

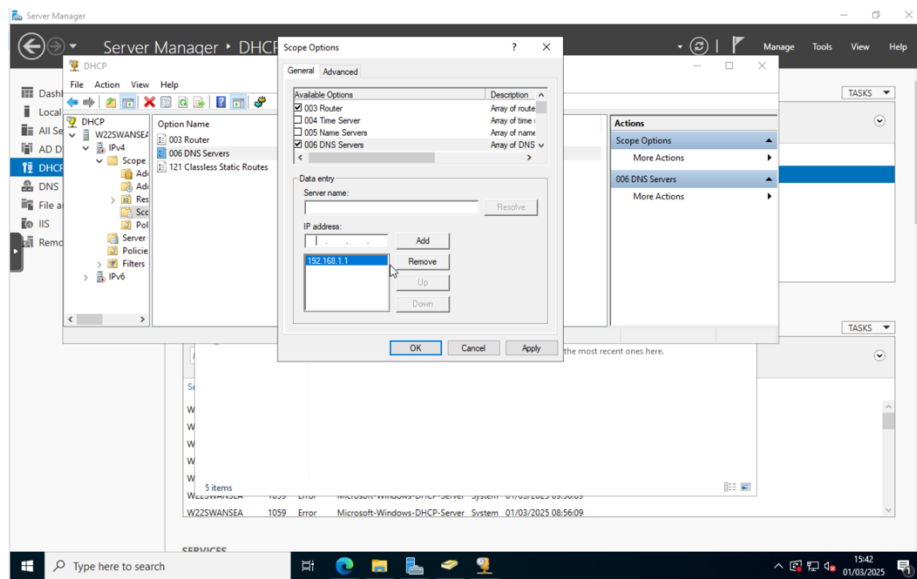
Afterwards, the unwanted hosts were deleted from the DNS server, leaving only the scopes for the configured interfaces.

Since Active Directory is based on a DNS system, one also needed to add known public DNSs as forwarders in the DNS Server's properties so that each client on the LAN would be able to resolve public domain names through Windows Server. Adding those as forwarders will prevent issues in the domain name resolution of the local forest that will be resolved first, and if not, then forwarded to public domain names.



21 - External DNSs added as forwarders in the DNS Server

To ensure clients on the LAN will be able to resolve the armand.acca domain, it was necessary to edit the previously assigned DNS server for the DHCP clients. So, in the DHCP Manager, the DNS for clients was changed from being 1.1.1.1 to 192.168.1.1 (WS IP).



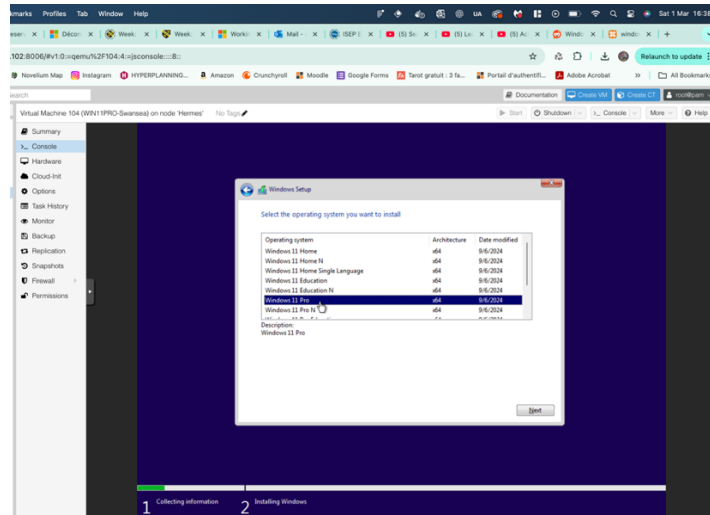
22 - Editing client's DNS in DHCP Manager

- Testing on Swansea's LAN -

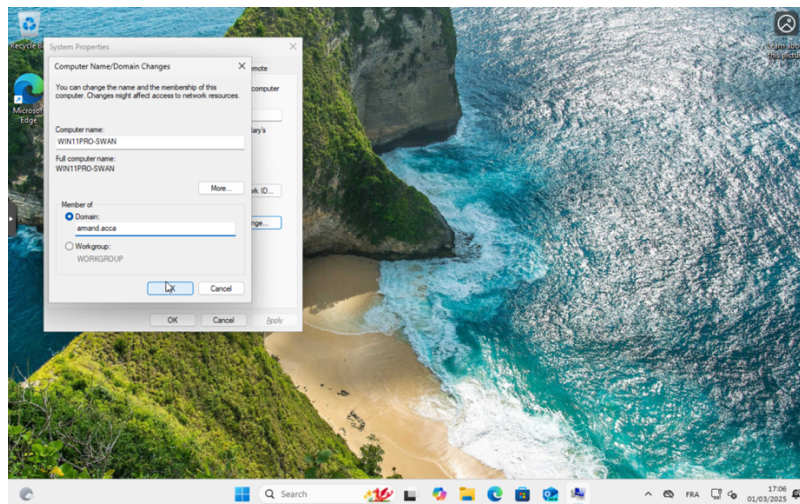
Before configuring the child domain and adding the users_list_updated.csv file, let's test the configuration. A test account with one's name was created (a.octeau) on the parent domain for testing purposes and will be used onwards.

However, an important issue quickly came in sight: the current installed version of Windows 11 on the client isn't a professional edition but a home one. Therefore, this VM couldn't join the parent domain. A new VM was then created from scratch with the right edition being installed. The firewall was configured again to allow for ICMPv4 pings.

Coursework 1 | NOSS

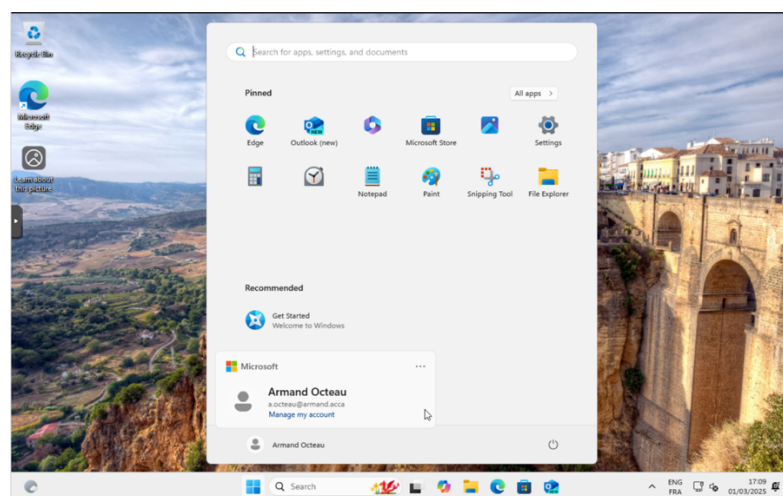


23 - Windows 11 Pro VM installation



24 - W11 VM being joined to the armand.acca forest

Now that this VM is joined to the newly created forest, let's try to connect to the a.octeau user previously created. And as it should, it works.



25 - Successful connection to a domain user in W11

Active Directory Configuration for Cardiff

- Child domain creation -

On Cardiff's LAN, the Debian server will be configured as a domain controller for the child domain **car.armand.acca** (standing for Cardiff).

On the kea-dhcp4-server configuration, the DNSs servers for Cardiff's clients were edited as being Debian's IP just like it was done on Swansea's LAN.

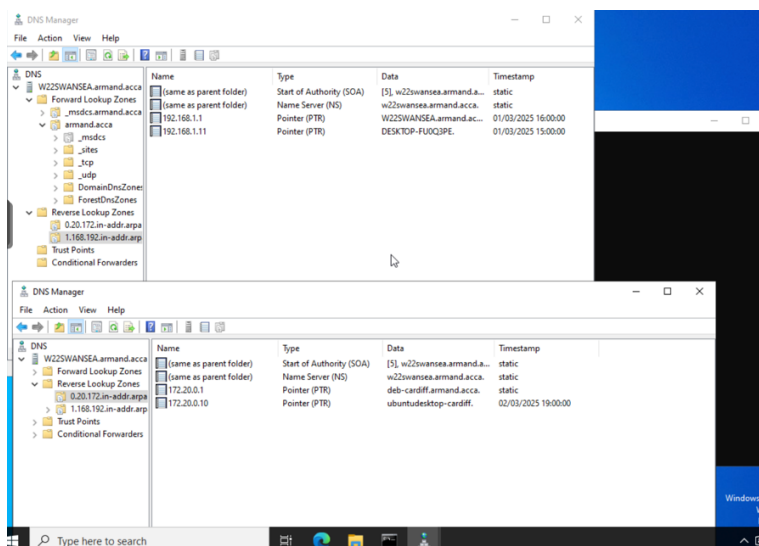
Required packages were also installed on Debian: samba (for AD), winbind (to link with Windows) and krb5-user (for Kerberos authentication), followed with other needed dependencies.

The /etc/resolv.conf file was edited as followed so Debian can resolve the domains.

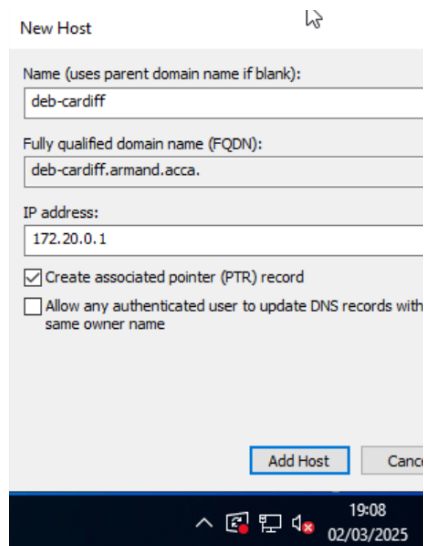
```
GNU nano 7.2
nameserver 192.168.1.1
search armand.acca
nameserver 172.20.0.1
search car.armand.acca
nameserver 1.1.1.1
nameserver 8.8.8.8
```

26 - /etc/resolv.conf file on Debian

On Windows Server's DNS, a new record was added for Debian's server being 172.20.0.1 alongside its PTR reverse lookup zone so that Swansea's clients could be able to connect to the remote AD.



27 - Reverse Lookup Zones on Windows Server's DNS



28 - New record for the Debian Server

The domain was provisioned with the samba-tool command, and then, for the configuration, it was set as the same as Windows. Samba was configured to listen only on the loopback and

LAN interfaces. IDmap ranges were also configured for the future newly created Unix users and the DNS forwarder was set as Windows Server's IP for external domain resolution on the Cardiff LAN.

```
[libdefaults]
    default_realm = CAR.ARMAND.ACCA
    dns_lookup_realm = false
    dns_lookup_kdc = true

[realms]
CAR.ARMAND.ACCA = {
    default_domain = car.armand.acca
    kdc=172.20.0.1
    admin_server=172.20.0.1
}
ARMAND.ACCA = {
    default_domain = armand.acca
    kdc=192.168.1.1
    admin_server=192.168.1.1
}

[domain_realm]
    .car.armand.acca = CAR.ARMAND.ACCA
    car.armand.acca = CAR.ARMAND.ACCA
    .armand.acca = ARMAND.ACCA
    armand.acca = ARMAND.ACCA

root@debiandserver-ca:~# date
Sat 8 Mar 02:42:52 GMT 2025
root@debiandserver-ca:~# _
```

29 - krb5.conf for Kerberos authentication on Debian

```
# Global parameters
[global]
    interfaces = 127.0.0.1 172.20.0.1
    bind interfaces only = yes
    dns forwarder = 192.168.1.1
    netbios name = debiandserver-ca
    realm = CAR.ARMAND.ACCA
    server role = active directory domain controller
    workgroup = CAR
    idmap_ldb:use rfc2307 = yes

    client signing = auto
    server signing = auto

    idmap config * : backend = tdb
    idmap config * : range = 3000-7999

    idmap config ARMAND:backend = ad
    idmap config ARMAND:schema_mode = rfc2307
    idmap config ARMAND:range = 10000-999999

    winbind use default domain = yes
    winbind enum users = yes
    winbind enum groups = yes

[sysvol]
    path = /var/lib/samba/sysvol
    read only = No

[netlogon]
    path = /var/lib/samba/sysvol/car.armand.acca/scripts
    read only = No

root@debiandserver-ca:~# date
Sat 8 Mar 02:43:24 GMT 2025
root@debiandserver-ca:~# _
```

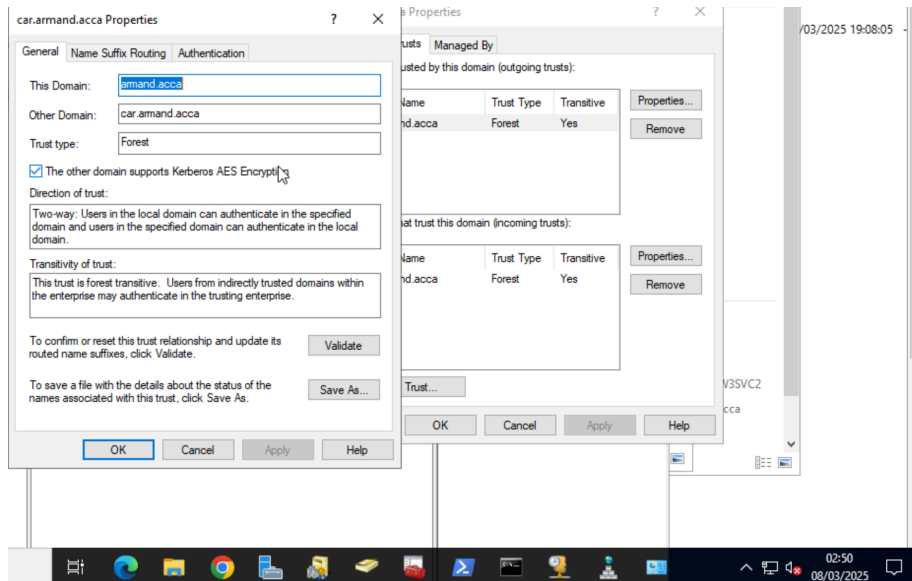
30 - Samba smb.conf configuration on Debian

Same as on Swansea's LAN, a test user was created for this course named a.music.

- Domain Trust -

To ensure the parent (armand.acca) and child (car.armand.acca) domains can communicate trustfully with each other, a trust certificate needed to be created for the two DCs. To do so, on Windows Server GUI, a new both-way trust was created as shown below.

This new certificate was also created on Debian's side automatically by providing Administrator credentials from the CAR domain to Windows. Trust was then successfully validated from both sides.



31 - Newly created domain trust on Windows Server

```

root@debianserver-ca:~# samba-tool domain trust validate armand.acca -U "Administrator@ARMAND.ACCA"
LocalDomain Netbios[CAR] DNS[car.armand.acca] SID[S-1-5-21-3852396905-4089760254-2723467934]
LocalTDO Netbios[ARMAND] DNS[armand.acca] SID[S-1-5-21-2260371945-2230544365-3860432542]
OK: LocalValidation: DC[\W22SHANSEA.armand.acca] CONNECTION[HERR_OK] TRUST[HERR_OK] VERIFY_STATUS_RETURNED
OK: LocalRediscover: DC[\W22SHANSEA.armand.acca] CONNECTION[HERR_OK]
RemoteDC Netbios[W22SHANSEA] DNS[W22SHANSEA.armand.acca] ServerType[PODC,GC,LDAP,DS,KDC,TIMESERV,CLOSEST,WRITABLE,GOOD_TIMESERV,FULL_SECRET_DOMAIN_6,ADS_WEB_SERVICE,DS_8,___unknown_00038000___]
Password for [Administrator@ARMAND.ACCA]:
OK: RemoteValidation: DC[\\debianserver-ca.car.armand.acca] CONNECTION[HERR_OK] TRUST[HERR_OK] VERIFY_STATUS_RETURNED
OK: RemoteRediscover: DC[\\debianserver-ca.car.armand.acca] CONNECTION[HERR_OK]
root@debianserver-ca:~# DATE
-bash: DATE: command not found
root@debianserver-ca:~# date
Sun 2 Mar 22:37:15 GMT 2025

```

32 - New domain trust validation on Debian side

- Testing on Cardiff's LAN -

Required packages were installed on the Ubuntu Desktop client (realmd, sssd, samba...) for configuration. The Ubuntu client was then joined to the CAR domain using the realm command and was also configured to allow the creation of a new home directory for each new user.

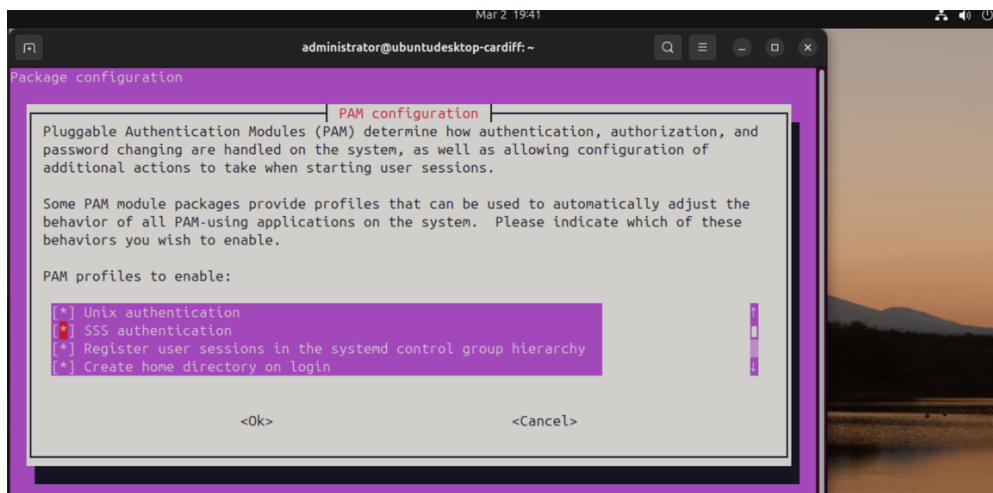
Tests users (ARMAND\a.octeau & CAR\a.music) were tested and the ubuntu client was successfully able to authenticate users from both the parent and child domain. A home directory was also created for both with correct permissions.

```

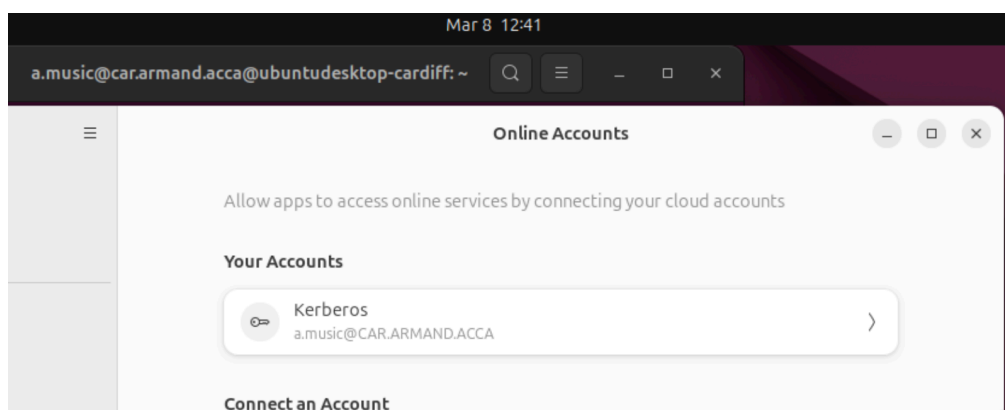
Mar 8 03:03
a.music@car.armand.acca@ubuntudesktop-cardiff:~$ realm join car.armand.acca
realm: Already joined to this domain
a.music@car.armand.acca@ubuntudesktop-cardiff:~$ realm list
car.armand.acca
  type: kerberos
  realm-name: CAR.ARMAND.ACCA
  domain-name: car.armand.acca
  configured: kerberos-member
  server-software: active-directory
  client-software: sssd
  required-package: sssd-tools
  required-package: sssd
  required-package: libnss-sss
  required-package: libpam-sss
  required-package: adcli
  required-package: samba-common-bin
  login-formats: %U@car.armand.acca
  login-policy: allow-realm-logins
a.music@car.armand.acca@ubuntudesktop-cardiff:~$ date
Sat Mar  8 03:03:09 AM GMT 2025
a.music@car.armand.acca@ubuntudesktop-cardiff:~$

```

33 - Successfully joined the child domain on Ubuntu Desktop



34 - PAM configuration on Cardiff's client



35 - Successfully connected child AD user on Ubuntu

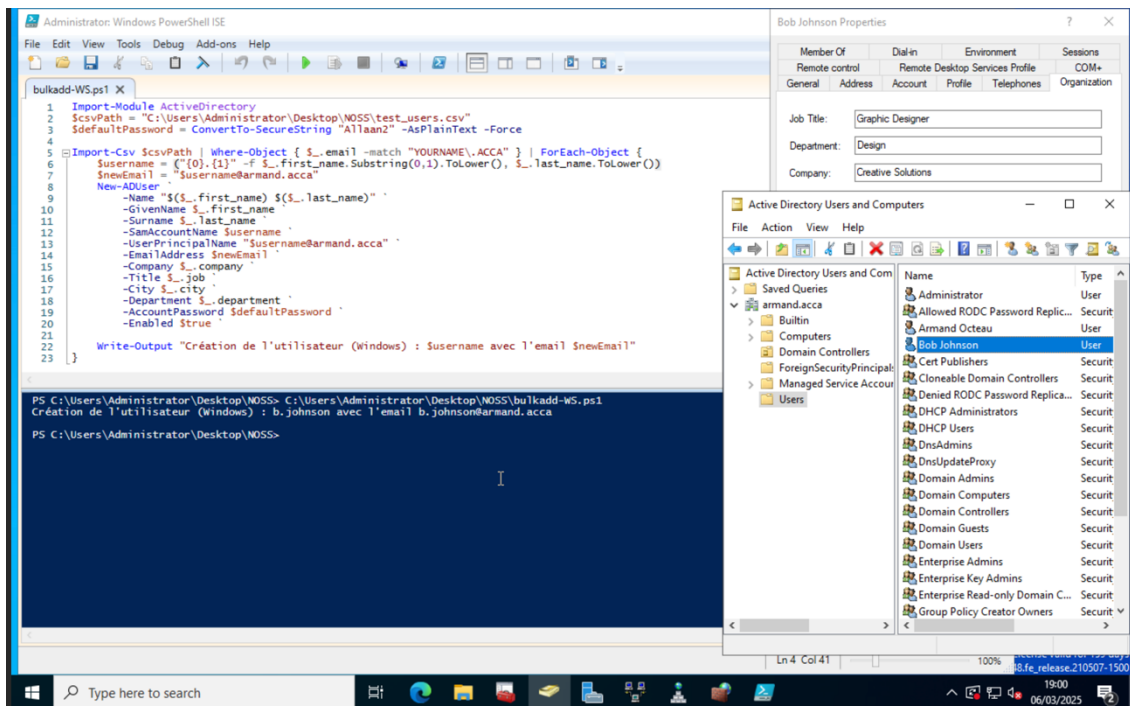
Importing AD Users

- On Swansea's DC -

To import the provided users_list_updated.csv file, it was necessary to understand it. Following this configuration, one chose to add users labelled as OLDNAME.ACCA to the child domain (Cardiff) and the ones labelled as YOURNAME.ACCA to the parent domain (Swansea).

To do so, ChatGPT was used to create a PowerShell script (ps1 file) to add the selected users to the parent domain on Swansea's DC. Users will be imported following this scheme: f.name for the username, following with its correct email username@domain and with its company, city and job information.

The scripts were uploaded to my personal web server so that the VMs only had to fetch an URL to get the files.



36 - PowerShell script to bulk import users on Windows Server

- On Cardiff's DC -

With samba-tool, it's not possible to create a user with the "company, job, city, etc." flags option. Therefore, the users that needed to be created on Debian would have those fields set as a description like shown on the following screenshot. Their password will be set the same for all the accounts just like we did on Windows Server.

```

GNU nano 7.2 bulkadd-DB.sh
#!/bin/bash
CSV_FILE="./users_list.csv"

tail -n +2 "$CSV_FILE" | while IFS=',' read -r first_name last_name company domain email job city department; do
    # Remove unnecessary quotes and spaces
    first_name=$(echo "$first_name" | tr -d '""' | sed 's/[[[:space:]]*//;s/[[[:space:]]*$//')
    last_name=$(echo "$last_name" | tr -d '""' | sed 's/[[[:space:]]*//;s/[[[:space:]]*$//')
    company=$(echo "$company" | tr -d '""' | sed 's/[[[:space:]]*//;s/[[[:space:]]*$//')
    domain=$(echo "$domain" | tr -d '""' | sed 's/[[[:space:]]*//;s/[[[:space:]]*$//')
    job=$(echo "$job" | tr -d '""' | sed 's/[[[:space:]]*//;s/[[[:space:]]*$//')
    city=$(echo "$city" | tr -d '""' | sed 's/[[[:space:]]*//;s/[[[:space:]]*$//')
    department=$(echo "$department" | tr -d '""' | sed 's/[[[:space:]]*//;s/[[[:space:]]*$//')

    # Skip users who do not have OLDNAME.ACCA as their domain
    [ "$domain" != "OLDNAME.ACCA" ] && continue

    firstchar=$(echo "$first_name" | cut -c1 | tr '[:upper:]' '[:lower:]')
    username="$firstchar.${last_name} | tr '[:upper:]' '[:lower:]'"
    newEmail="$username@car.armand.acca"

    samba-tool user create "$username" "Allaan2" \
        --given-name="$first_name" \
        --surname="$last_name" \
        --mail-address="$newEmail" \
        --description="Company: $company, Job: $job, City: $city, Department: $department"

    echo "Création de l'utilisateur (Samba) : $username avec l'email $newEmail"
done

```

37 -Bash script to bulk import users on Debian with samba-tool

For testing, a small csv file of two users was created: one user with the OLDNAME.ACCA domain and one with YOURNAME.ACCA. As expected, only the right user was added to the AD database for each the parent and child domain.

first_name	last_name	company	domain	email	job	city	department
Alice	Smith	Tech Innovators Inc	OLDNAME.ACCA	alice.smith@OLDNAME.ACCA	Senior Developer	San Francisco	Engineering
Bob	Johnson	Creative Solutions	YOURNAME.ACCA	bob.johnson@YOURNAME.ACCA	Graphic Designer	Los Angeles	Design

38 - A two user small csv file for testing purposes

```

root@debianserver-ca:~# sh bulkadd-DB.sh
User 'a.smith' added successfully
Création de l'utilisateur (Samba) : a.smith avec l'email a.smith@car.armand.acca
root@debianserver-ca:~# wbinfo -u
CAR\administrator
CAR\guest
CAR\krbtgt
CAR\a.music
CAR\a.smith
root@debianserver-ca:~# DATE
-bash: DATE: command not found
root@debianserver-ca:~# date
Thu 6 Mar 19:04:26 GMT 2025
root@debianserver-ca:~#

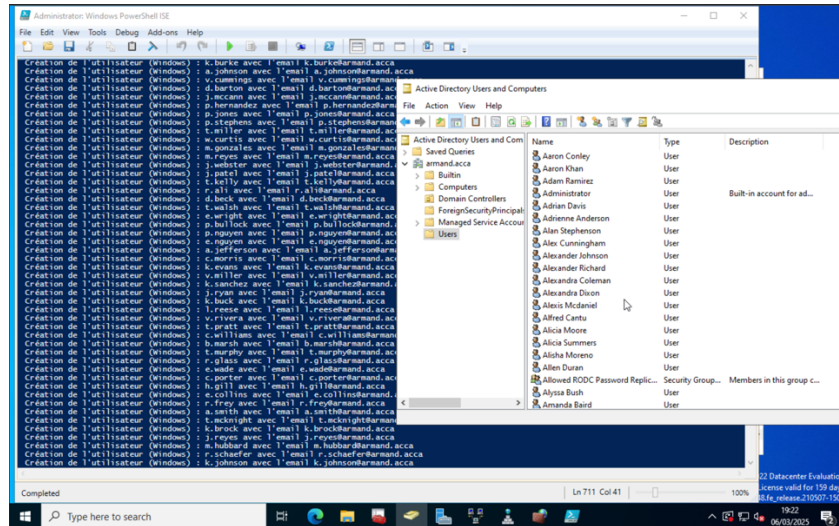
```

39 – Output on Debian after executing the script on the test_users.csv file

Coursework 1 | NOSS

Now that the script is working, we'll import the complete database file.

N.B: Since the provided users_list_updated.csv file is AI generated, some users will be duplicated and therefor not added to the AD domain.

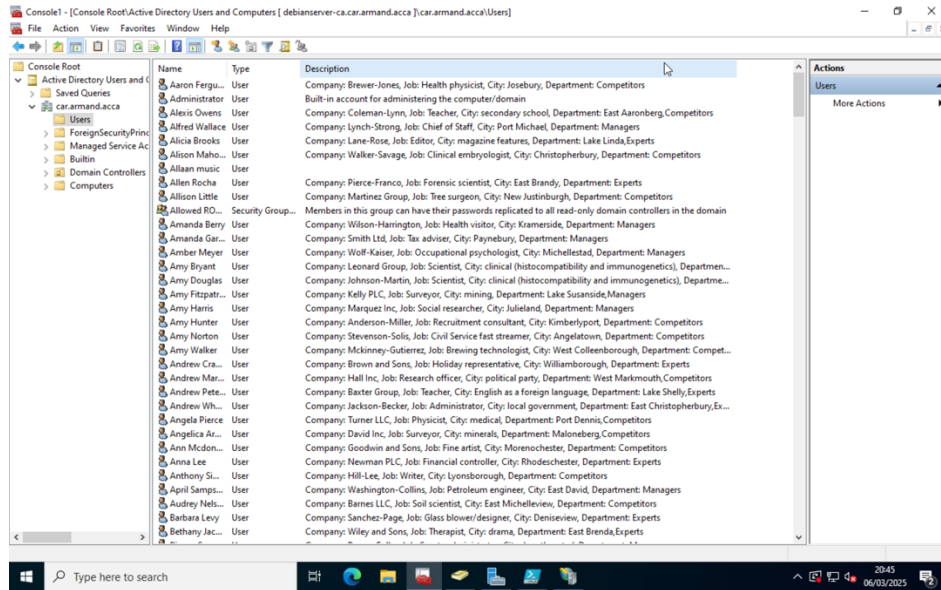


40 - Adding users to the parent AD domain

```
Création de l'utilisateur (Samba) : m.davis avec l'email m.davis@car.armand.acca
User 't.fitzgerald' added successfully
Création de l'utilisateur (Samba) : t.fitzgerald avec l'email t.fitzgerald@car.armand.acca
User 'g.boone' added successfully
Création de l'utilisateur (Samba) : g.boone avec l'email g.boone@car.armand.acca
User 'h.chang' added successfully
Création de l'utilisateur (Samba) : h.chang avec l'email h.chang@car.armand.acca
User 'j.watson' added successfully
Création de l'utilisateur (Samba) : j.watson avec l'email j.watson@car.armand.acca
User 'n.johnson' added successfully
Création de l'utilisateur (Samba) : n.johnson avec l'email n.johnson@car.armand.acca
ERROR(lsb): Failed to add user 'a.martinez': - samldb: samAccountName 'a.martinez' already in use!
User 'a.martinez' added successfully
Création de l'utilisateur (Samba) : a.martinez avec l'email a.martinez@car.armand.acca
User 'm.jenkins' added successfully
Création de l'utilisateur (Samba) : m.jenkins avec l'email m.jenkins@car.armand.acca
User 'w.hodge' added successfully
Création de l'utilisateur (Samba) : w.hodge avec l'email w.hodge@car.armand.acca
User 'd.wright' added successfully
Création de l'utilisateur (Samba) : d.wright avec l'email d.wright@car.armand.acca
User 'k.williams' added successfully
Création de l'utilisateur (Samba) : k.williams avec l'email k.williams@car.armand.acca
User 'n.nolan' added successfully
Création de l'utilisateur (Samba) : n.nolan avec l'email n.nolan@car.armand.acca
User 'l.nelson' added successfully
Création de l'utilisateur (Samba) : l.nelson avec l'email l.nelson@car.armand.acca
User 'd.thomas' added successfully
Création de l'utilisateur (Samba) : d.thomas avec l'email d.thomas@car.armand.acca
User 'r.griffin' added successfully
Création de l'utilisateur (Samba) : r.griffin avec l'email r.griffin@car.armand.acca
ERROR(lsb): Failed to add user 's.clark': - samldb: samAccountName 's.clark' already in use!
User 's.clark' added successfully
Création de l'utilisateur (Samba) : s.clark avec l'email s.clark@car.armand.acca
User 'g.rogers' added successfully
Création de l'utilisateur (Samba) : g.rogers avec l'email g.rogers@car.armand.acca
User 'l.schwartz' added successfully
Création de l'utilisateur (Samba) : l.schwartz avec l'email l.schwartz@car.armand.acca
User 'g.callahan' added successfully
Création de l'utilisateur (Samba) : g.callahan avec l'email g.callahan@car.armand.acca
User 'j.kelly' added successfully
Création de l'utilisateur (Samba) : j.kelly avec l'email j.kelly@car.armand.acca
User 'e.hernandez' added successfully
Création de l'utilisateur (Samba) : e.hernandez avec l'email e.hernandez@car.armand.acca
User 'j.mccormick' added successfully
Création de l'utilisateur (Samba) : j.mccormick avec l'email j.mccormick@car.armand.acca
User 'b.serrano' added successfully
Création de l'utilisateur (Samba) : b.serrano avec l'email b.serrano@car.armand.acca
root@debianserver-ca:~# date
Thu 6 Mar 19:33:35 GMT 2025
root@debianserver-ca:~# _
```

41 - Adding users to the child AD domain

Coursework 1 | NOSS

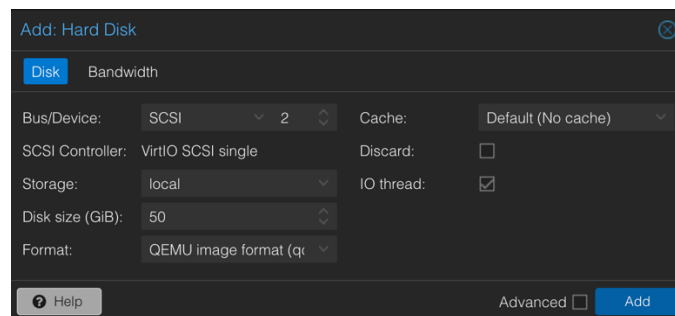


42 - Child domain accessed with a remote console on WS showing the added imported users

Shared Folder on Swansea's

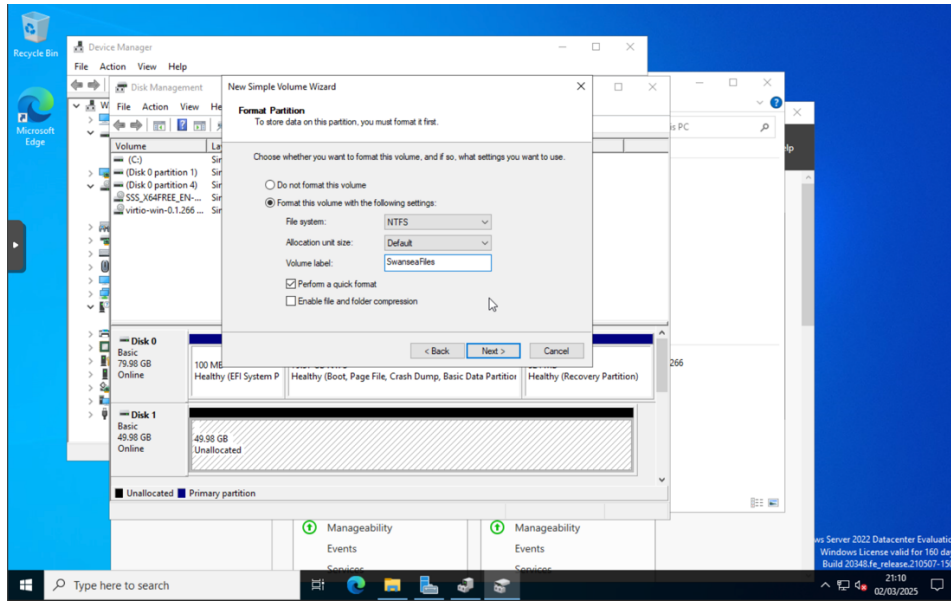
- *Creating the new disk* -

On Proxmox, a new virtual disk of 50Gb was created and virtually plugged in the Windows Server VM.



43 - New disk creation wizard on Proxmox

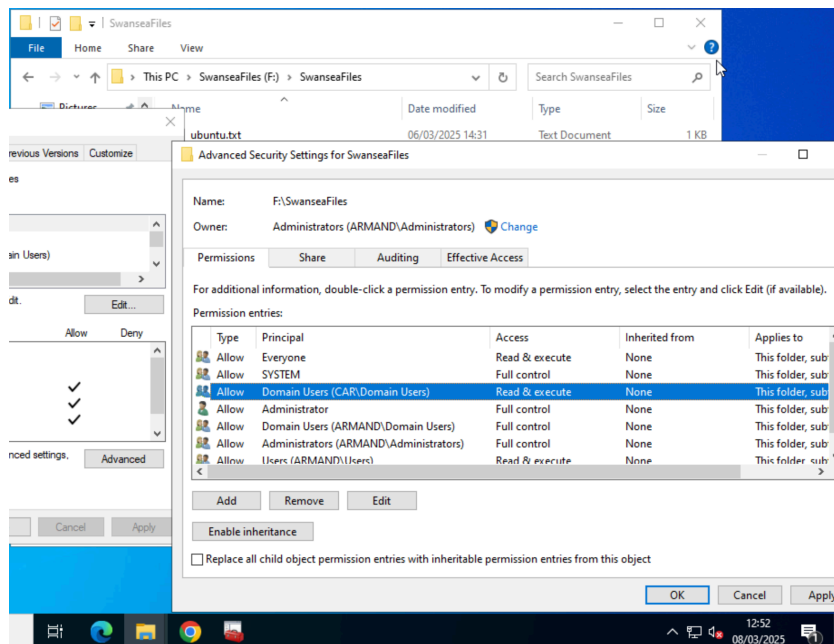
The disk was then formatted on Windows after a restart to a NTFS file system. It was then mounted as "SwanseaFiles" to the F:/ locator.



44 – Formating and mounting the new disk on Windows Server

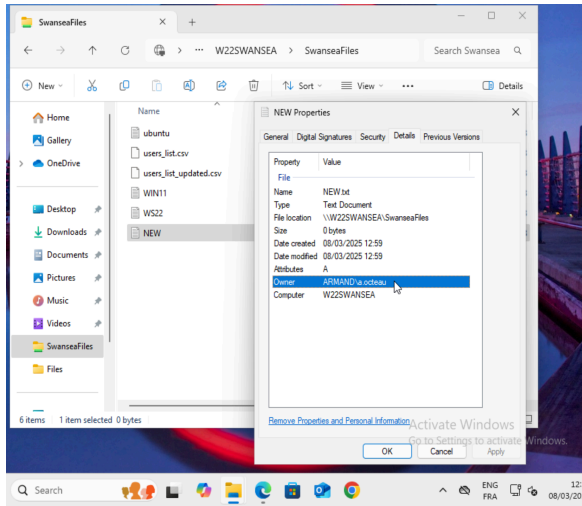
- Setting permissions -

Afterwards, the newly created folder “SwanseaFiles” inside the F:/ disk carried by the same name was shared and the correct permissions were set on its properties. Domain Users of the parent armand.acca have full control permissions (write, read, execute...) applied to folders and subfolders. And Domain Users of the child car.armand.acca domain have only read & execute permissions (as well as Everyone and ARMAND\Users to prevent issues and unwanted access).

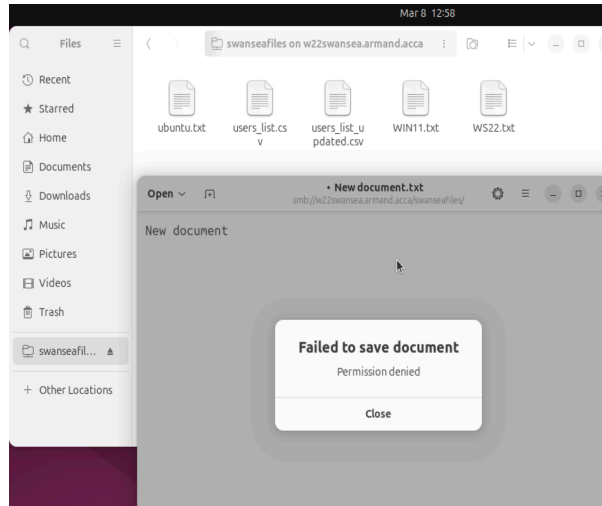


45 - Permissions for SwanseaFiles

- Testing -



46 - Can successfully add files to shared folders in W11 client with a ARMAND.ACCA user

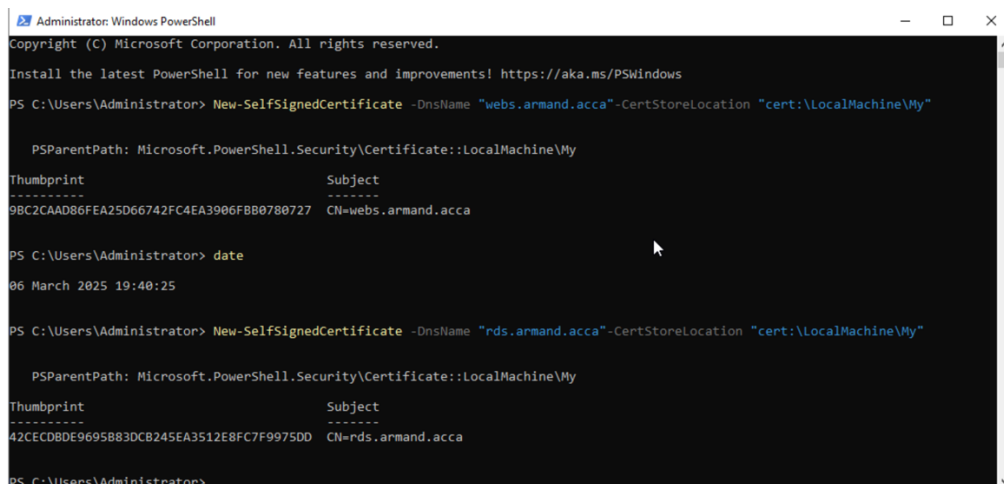


47 - Permission denied for a child domain user

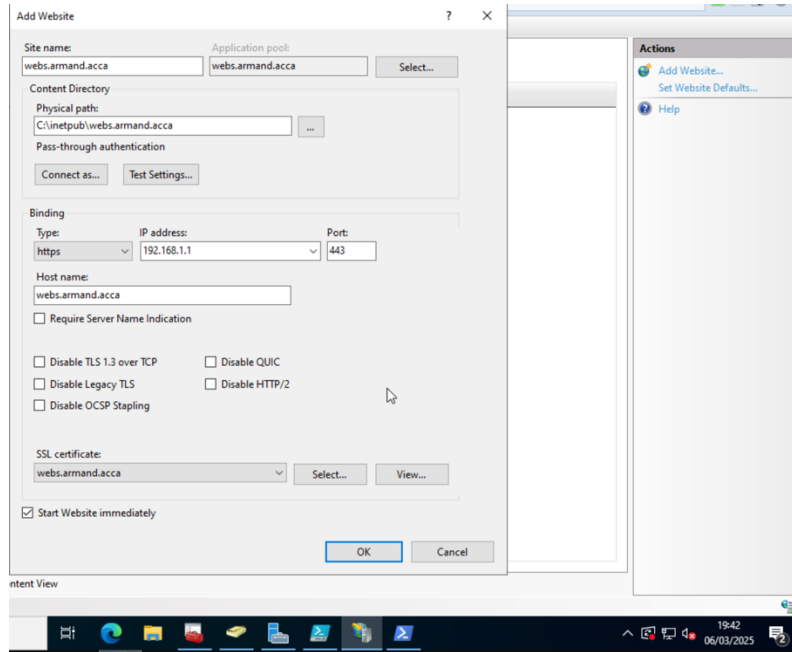
Configuring IIS's server

- Configuring -

On Windows Server, the IIS role and feature was installed alongside its dependencies. During this coursework, we'll need to configure two websites: one with a "Welcome to Swansea" page and one with the RDS control panel. Therefore, two self-issued SSL certificates needed to be created for both child domains "webs.armand.acca" & "rds.armand.acca".



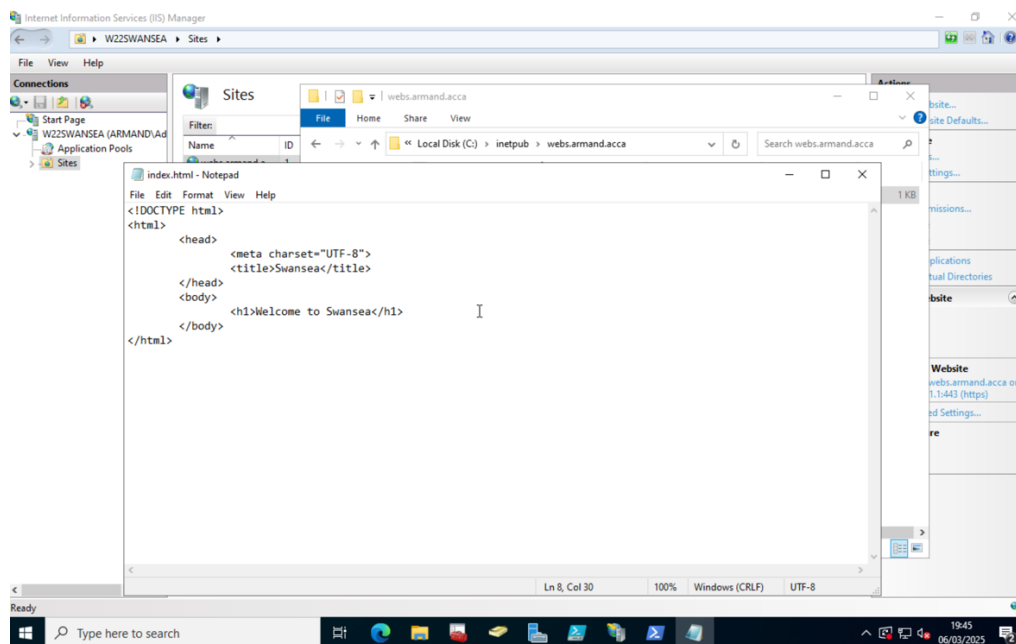
48 - Self generating two SSL certificates for both respective child domains



49 - Creating the webs.armand.acca website on IIS

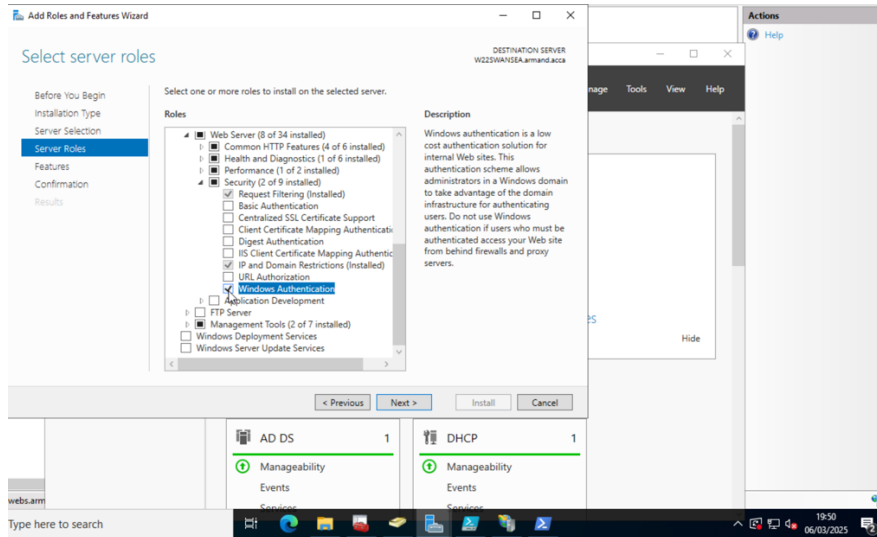
The website was then created and modified in its bindings options to provide it with the previously generated SSL certificate and enabling it only on https (port 443).

A small basic html file was created and placed in the root directory of the website.

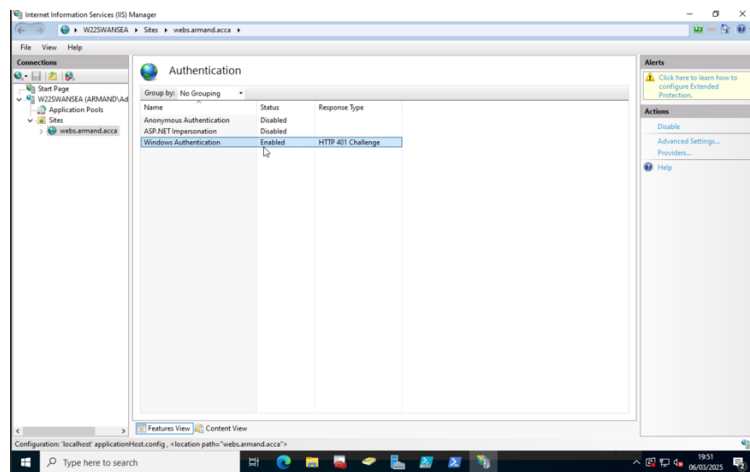


50 - Welcome to Swansea HTML file

It was needed to install the Windows Authentication feature of IIS that wasn't previously installed.



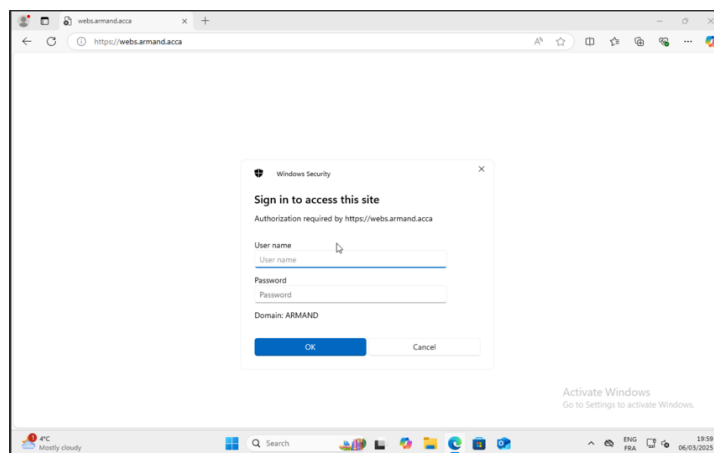
51 - Adding the Windows Authentication feature



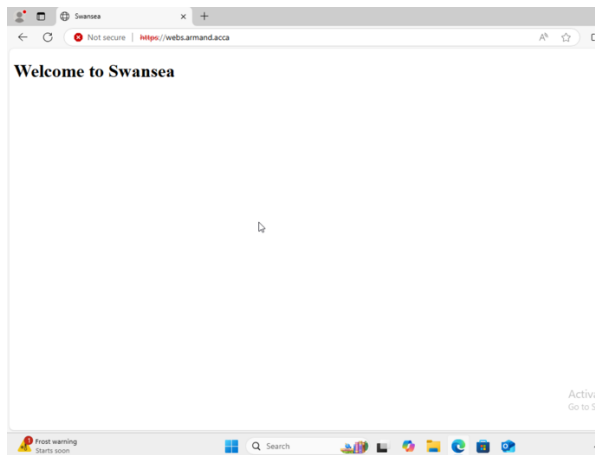
52 - Enabling Windows Authentication on the webs.armand.acca website

Furthermore, a new scope in the DNS server had to be created in order for the websites to be resolvable.

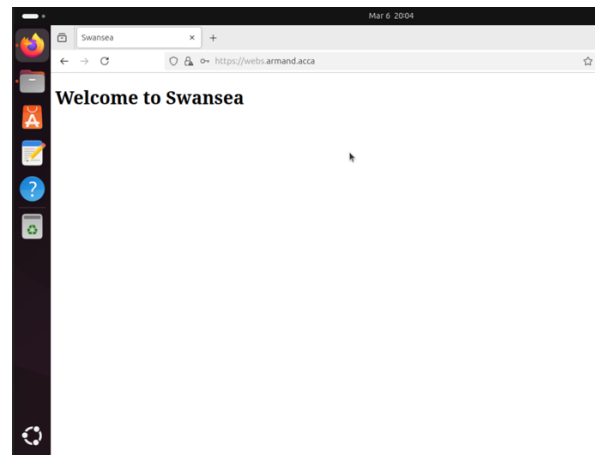
- Testing -



53 - Windows Authentication prompt at website opening



54 - webs.armand.acca from Swansea's client



55 - webs.armand.acca from Cardiff's client

- Issue Resolving -

Due to local region rules, the websites weren't accessible on localhost. The Windows authentication failed. Therefore, the following rule was modified to allow connection from the localhost (being Windows Server).

```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\Administrator> reg add "HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Control\Lsa" /v DisableLoopbackCheck /t REG_DWORD /d 1 /f
The operation completed successfully.
PS C:\Users\Administrator> iisreset
Attempting stop...
Internet services successfully stopped
Attempting start...
Internet services successfully restarted
PS C:\Users\Administrator> nslookup webs.armand.acca
Server: UnKnown
Address: ::1

Name: webs.armand.acca
Address: 192.168.1.1

PS C:\Users\Administrator> date
07 March 2025 13:06:03

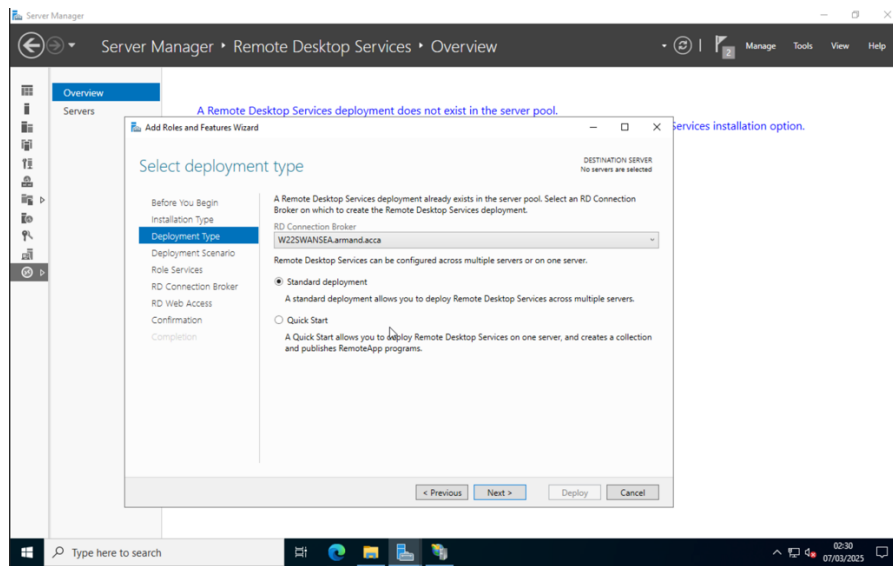
PS C:\Users\Administrator> _
```

56 - Authorizing localhost connection for IIS websites

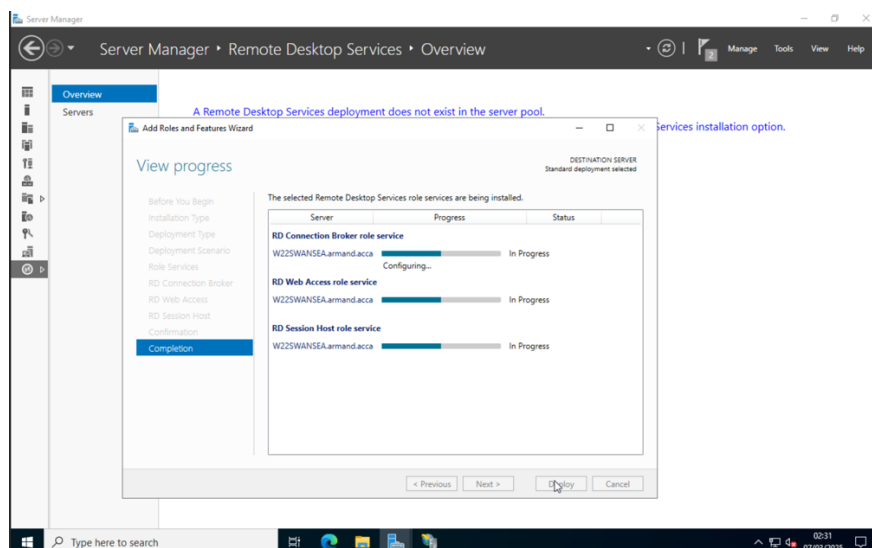
Remote Desktop Services

- Configuring -

Using the RDS option in the “add roles and features” wizard, the necessary features were added to Windows Server (Remote Web Access, Remote Gateway & Remote Session Host alongside IIS dependencies). A few restarts and reinstallation were needed due to issues on the host machine. Installation would fail constantly. After a few tries, it finally succeeded.



57 - RDS installation wizard



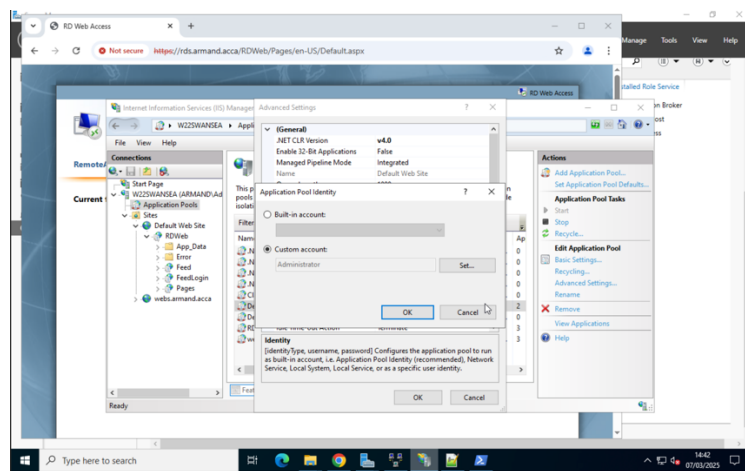
58 - RDS configuration by the wizard

On IIS, it was needed to add manually the RDWeb application pool since it wasn't installed properly. In the binding settings, the Default Web Site was configured the same as previously but for the rds.armand.acca domain, giving it its previously generated SSL certificate.

An HTTP redirect was also configured so that queries to the home page would automatically be redirected to /RDWeb/.

- Issue Resolving -

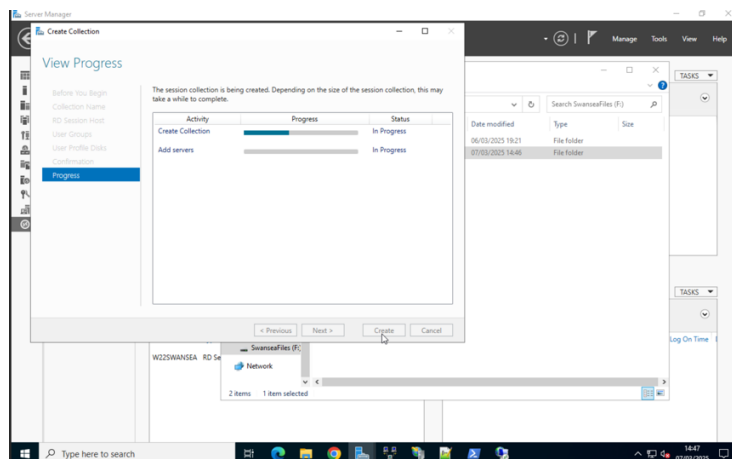
When accessing the website, numerous errors were encountered. The most difficult to resolve was the access denied error to the Temporary folder of Microsoft .NET. This issue was solved by specifying the access user as “Administrator” in the RDWeb app configuration as shown below.



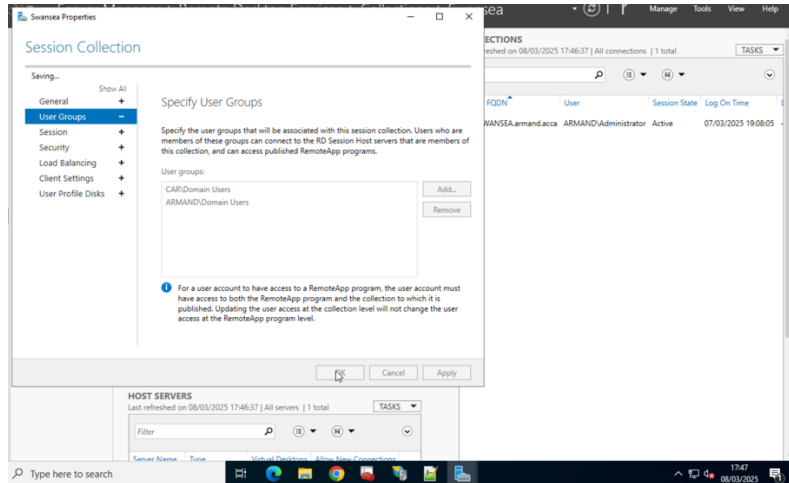
59 - Setting Administrator as the user to use in RDWeb's settings

- Creating the collection -

In the server manager, a new collection for RDS was created. User Groups were configured as “CAR\Domain Users” and “ARMAND\Domain Users” to grant access to both domains members.

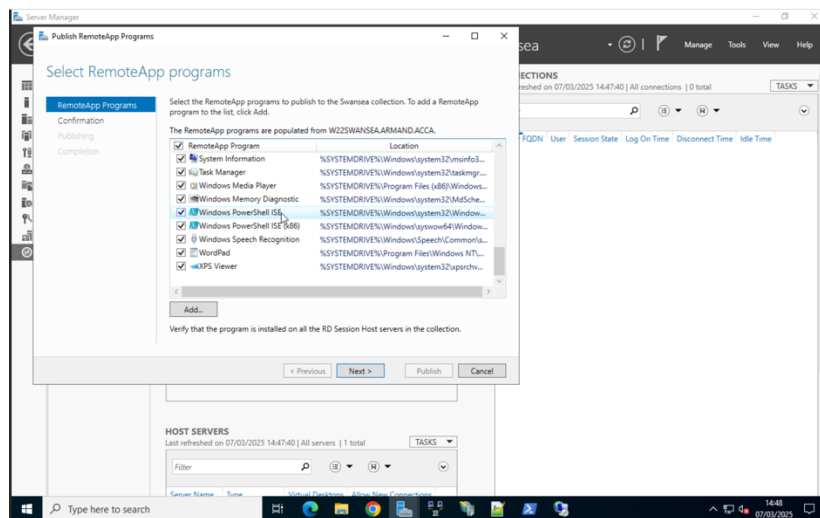


60 - Creating Swansea's collection for RDS



61 - Adding groups for the collection access

Remote applications were then added to the collection and selected so that they'll be shown on the RDS webpage.

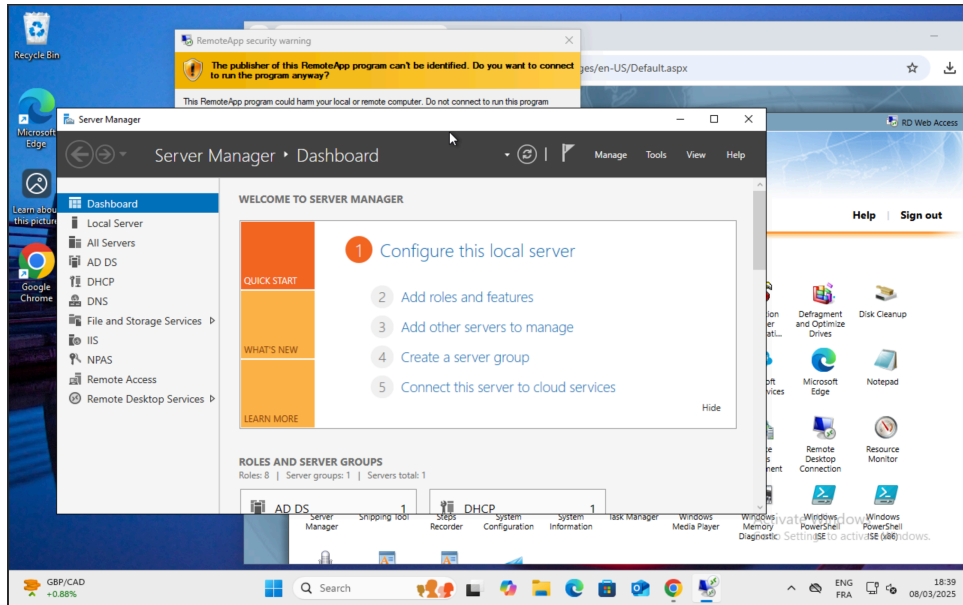


62 - Adding applications to the collection

- Testing -

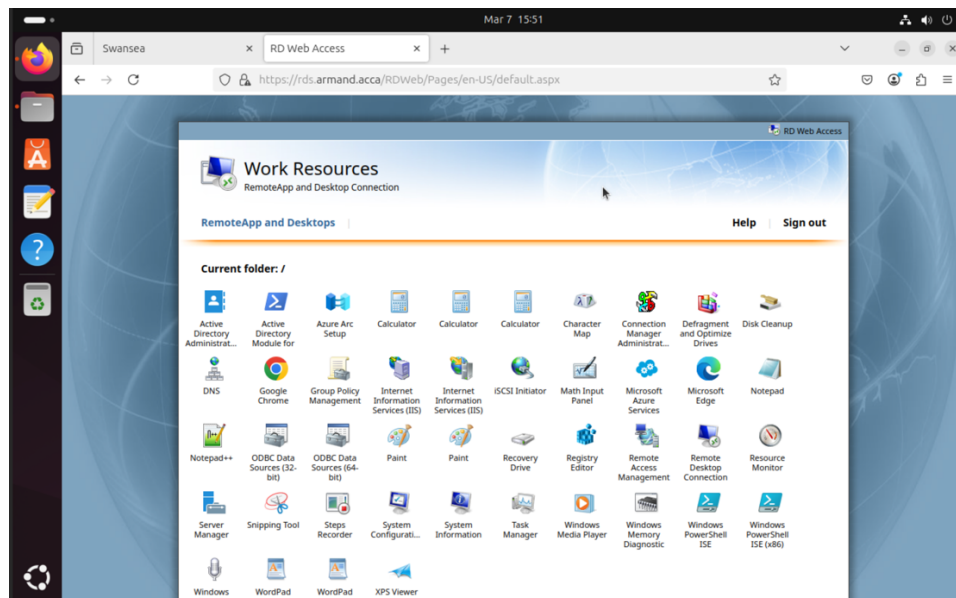
Let's try connecting to Windows Server on the Swansea's LAN with a parent domain user and launching a remote app session. The chosen app will be Server Manager.

Coursework 1 | NOSS



63 - Connecting to the Server Manager from RDS on Windows 11

As expected, the session authenticates, the remote app launches and everything is working as it should. The website is also accessible to connect on the child domain in Cardiff as shown below.



64 - Connecting to the rds.armand.acca website from Cardiff with a child domain user

This coursework is now fully completed, and the network configuration is fully operational and working.