**Research & Development Document**

**Title: VPN Setup Using Hyper-V – Point-to-Site and Site-to-Site Configuration**

**Section 1: Point-to-Site (P2S) VPN Setup**

**To** Enable remote client machines to securely connect to a private network (hosted on-premises or in Azure) via a VPN tunnel.

**🔧 Technologies Used:**

* Windows Server (2016/2019/2022)
* Hyper-V Virtual Machines
* RRAS (Routing and Remote Access Service)
* Self-signed Root Certificate / Enterprise CA
* PowerShell
* Azure (Optional, if connecting to Azure VNet)

**🔄 Architecture Overview:**

rust

CopyEdit

Client Device (Laptop) ---> VPN Tunnel ---> RRAS Server (Host VM on Hyper-V) ---> Internal Network

**📋 Prerequisites:**

* Hyper-V enabled on the host machine
* At least one VM running Windows Server with 2 NICs (External and Internal)
* Static IP configuration on internal network adapter
* Domain or Workgroup setup

**🛠 Steps to Configure Point-to-Site VPN**

**1. Set Up RRAS on the Hyper-V VM**

powershell

CopyEdit

Install-WindowsFeature RemoteAccess, RSAT-RemoteAccess, DirectAccess-VPN, Routing -IncludeManagementTools

* Open Routing and Remote Access from Server Manager.
* Configure and Enable Routing and Remote Access.
  + Choose "Custom Configuration" → "VPN access" → "NAT routing" if needed.
  + Start the RRAS service.

**2. Configure VPN with a Certificate**

* Create a **self-signed root certificate** on the RRAS server or use enterprise CA.

powershell

CopyEdit

$cert = New-SelfSignedCertificate -Type Custom -KeySpec Signature `

-Subject "CN=MyVPNRootCert" -KeyExportPolicy Exportable `

-HashAlgorithm sha256 -KeyLength 2048 -CertStoreLocation "Cert:\CurrentUser\My" `

-KeyUsageProperty Sign -KeyUsage CertSign

* Export the certificate and install it on client machines under **Trusted Root CA**.

**3. Configure User Access**

* Ensure VPN users have dial-in access enabled.
* Optionally create VPN-specific users in Active Directory or local accounts.

**4. Client Configuration**

* Export a connection .pbk file or manually configure VPN on Windows client:
  + Type: **IKEv2** or **SSTP**
  + Use public IP or DNS of RRAS server
  + Use user credentials configured earlier

**5. Test VPN**

* From client machine, connect via VPN.
* Try accessing internal network resources or ping internal servers.

**🔎 Notes:**

* For Azure P2S: You must configure the Azure VPN Gateway for P2S using certificates or RADIUS.
* Consider using NPS (Network Policy Server) for advanced authentication control.

**Section 2: Site-to-Site (S2S) VPN Setup Using Hyper-V**

**✅ Objective:**

Establish a permanent tunnel between two sites (e.g., Head Office and Branch Office), each running on Hyper-V environments.

**🔧 Technologies Used:**

* Windows Server with RRAS on both ends
* Hyper-V for hosting virtualized infrastructure
* Static Public IPs or DDNS for dynamic IPs
* IPSec/IKEv2 for secure tunnels

**🔄 Architecture Overview:**

css

CopyEdit

[Site A Network] <--> RRAS Server A <-- S2S Tunnel --> RRAS Server B <--> [Site B Network]

**📋 Prerequisites:**

* Two Hyper-V VMs with Windows Server (2016 or later) installed (one per site)
* Each VM must have two NICs (internal + external)
* Static IP addresses for both ends (or DDNS)
* Firewall configured to allow VPN traffic (UDP 500, UDP 4500, ESP)

**🛠 Steps to Configure Site-to-Site VPN**

**1. Install RRAS on Both Servers**

powershell

CopyEdit

Install-WindowsFeature RemoteAccess, RSAT-RemoteAccess, DirectAccess-VPN, Routing -IncludeManagementTools

* Use Server Manager → Routing and Remote Access → Configure RRAS
* Choose “Secure connection between two private networks”

**2. Configure Static Routes and Interfaces**

* Define **Demand-dial interface** for the remote site
* Use static IP address or DNS of the remote RRAS server
* Enter preshared key (must match on both ends)
* Define **IP addressing** and **routes** to the remote subnet

**3. Create Static Route to Remote Subnet**

powershell

CopyEdit

route add 192.168.20.0 mask 255.255.255.0 192.168.10.1

**4. Configure Firewall and NAT if needed**

* Open required ports:
  + UDP 500 (ISAKMP)
  + UDP 4500 (NAT-T)
  + ESP (IP Protocol 50)
* If using NAT, configure NAT rules on RRAS

**5. Test S2S Tunnel**

* Ping remote subnet from either end.
* Check tunnel status via RRAS console or Get-VpnS2SInterface

**💡 Optional: Use Windows PowerShell to Automate**

powershell

CopyEdit

Add-VpnS2SInterface -Name "BranchVPN" -Destination "203.0.113.1" -Protocol IKEv2 `

-AuthenticationMethod PSKOnly -SharedSecret "YourSecretKey" `

-IPv4Subnet @("192.168.20.0/24") -EnableQoS $false

**📌 Troubleshooting**

| **Issue** | **Possible Cause** | **Solution** |
| --- | --- | --- |
| VPN not connecting | Firewall blocking ports | Allow UDP 500, 4500, and IPsec |
| Tunnel up, but no traffic | Missing routes or NAT misconfig | Check route tables and NAT configuration |
| P2S users can't access resources | Incorrect DNS or certificate | Ensure client uses correct root cert and DNS |