

Active Directory DNS Security Hardening Guide

Important Notes:

- Replace `domain.local` with YOUR actual domain name (e.g., `company.com`, `contoso.local`)
 - Replace `10.0.0.2` and `10.0.0.3` with YOUR actual secondary DNS server IP addresses
 - Replace IP ranges like `10.0.0.0/8` with YOUR actual internal network ranges
 - When you see "domain controller" mentioned, use YOUR domain controller's hostname
-

1. Enable DNSSEC (DNS Security Extensions)

DNSSEC provides authentication and integrity verification for DNS responses, preventing DNS spoofing and cache poisoning attacks.

Enable DNSSEC on a Zone (Server Manager Method):

1. Open **Server Manager** > **Tools** > **DNS**
2. Expand your DNS server, then expand **Forward Lookup Zones**
3. Right-click your domain zone (e.g., `domain.local`) - use **YOUR domain name** and select **DNSSEC** > **Sign the Zone**
4. In the Zone Signing Wizard, click **Next**
5. Select **Customize zone signing parameters** and click **Next**
6. Configure Key Signing Key (KSK):
 - Click **Add** under Key Master
 - Select your domain controller (your server's name will appear here)
 - Set cryptographic algorithm to **RSA/SHA-256**
 - Key length: **2048 bits**
 - Click **OK**
7. Configure Zone Signing Key (ZSK):
 - Use similar settings as KSK
 - Enable **Enable automatic rollover**
8. Click **Next** through the remaining options and **Finish**

Validate DNSSEC (PowerShell - Recommended for Verification):

```
powershell
```

```
# Replace "domain.local" with YOUR domain name
```

```
Get-DnsServerDnsSecZoneSetting -ZoneName "domain.local"
```

```
# Verify DNSSEC signatures
```

```
Resolve-DnsName -Name "domain.local" -Type DNSKEY -Server localhost
```

2. Configure Secure Dynamic Updates

Prevent unauthorized clients from registering or updating DNS records.

Configure Secure Dynamic Updates:

1. Open **Server Manager** > **Tools** > **DNS**
2. Expand **Forward Lookup Zones**
3. Right-click your zone (e.g., `domain.local`) - use **YOUR domain name**) and select **Properties**
4. On the **General** tab, set **Dynamic updates** dropdown to **Secure only**
5. Click **OK**

Restrict DNS Updates to Domain Members Only:

1. In DNS Manager, right-click the zone and select **Properties**
 2. Go to the **Security** tab
 3. If **Authenticated Users** is present, select it and click **Remove**
 4. Click **Add**
 5. Click **Object Types**, check **Groups**, click **OK**
 6. Type **Domain Computers** and click **Check Names**, then **OK**
 7. Repeat to add **Domain Controllers** group
 8. For each group added, select it and check:
 - **Create all child objects**
 - **Delete all child objects**
 9. Click **OK**
-

3. Enable DNS Scavenging

Remove stale DNS records automatically to prevent DNS pollution and potential security risks.

Configure DNS Scavenging:

1. Open **Server Manager > Tools > DNS**
2. Right-click the DNS server name (your server's hostname at the top) and select **Set Aging/Scavenging for All Zones**
3. Check **Scavenge stale resource records**
4. Set **No-refresh interval: 7 days**
5. Set **Refresh interval: 7 days**
6. Click **OK**
7. In the confirmation dialog, select **Apply these settings to existing Active Directory-integrated zones**
8. Click **OK**

Enable Automatic Scavenging on the Server:

1. Right-click the DNS server name (your server's hostname) and select **Properties**
 2. Go to the **Advanced** tab
 3. Check **Enable automatic scavenging of stale records**
 4. Set scavenging period: **7 days**
 5. Click **OK**
-

4. Implement DNS Cache Locking

Prevent DNS cache poisoning by locking cached DNS records.

Configure DNS Cache Locking (PowerShell Only - No GUI Available):

```
powershell
```

```
# Set cache locking to 100% (maximum security)
# This means cached records cannot be overwritten until cache lifetime expires
Set-DnsServerCache -LockingPercent 100

# Verify setting
Get-DnsServerCache | Select LockingPercent
```

Recommended value: 100 (records cannot be overwritten until TTL expires)

5. Enable DNS Query Logging

Monitor DNS queries to detect suspicious activity and potential attacks.

Enable Debug Logging:

1. Open **Server Manager > Tools > DNS**
2. Right-click the DNS server (your server's hostname) and select **Properties**
3. Go to the **Debug Logging** tab
4. Check **Log packets for debugging**
5. Select the following options:
 - ✓ **Outgoing**
 - ✓ **Incoming**
 - ✓ **Queries/Transfers**
 - ✓ **Updates**
 - ✓ **Notifications**
 - ✓ **Questions**
 - ✓ **Answers**
6. Under **Log file**, ensure the path is: `%SystemRoot%\System32\dns\dns.log`
7. Set **Maximum size (bytes)**: `500000000` (500 MB)
8. Click **OK**

Enable Analytic Logging (PowerShell - Recommended for Better Performance):

```
powershell
```

```
# Enable DNS analytic logging (less performance impact than debug logging)
wevtutil sl "Microsoft-Windows-DNSServer/Analytical" /e:true

# View recent DNS queries
Get-WinEvent -LogName "Microsoft-Windows-DNSServer/Analytical" -MaxEvents 100 | Format-Table TimeCreated, Message
```

6. Configure DNS Response Rate Limiting (RRL)

Mitigate DNS amplification attacks by limiting response rates to suspicious queries.

Note: RRL is only available on Windows Server 2016 and later. No GUI available - must use PowerShell.

Enable RRL (PowerShell Only):

```
powershell

# Enable Response Rate Limiting
Set-DnsServerResponseRateLimiting -Mode Enable -Force

# Configure RRL parameters (use defaults or customize)
Set-DnsServerResponseRateLimiting `
    -ResponsesPerSec 5 `
    -ErrorsPerSec 5 `
    -WindowInSec 5 `
    -IPv4PrefixLength 24 `
    -IPv6PrefixLength 56 `
    -LeakRate 3 `
    -TruncateRate 2

# Verify RRL settings
Get-DnsServerResponseRateLimiting
```

Parameters Explained:

- **ResponsesPerSec:** Maximum identical responses per second (default: 5)
- **ErrorsPerSec:** Maximum error responses per second (default: 5)
- **LeakRate:** Ratio of responses still sent to clients (3 = 1 in 3 responses allowed)
- **TruncateRate:** Ratio of truncated responses (forces client to retry with TCP)

7. Configure DNS Recursion Settings

Control how your DNS server handles recursive queries.

Disable Recursion (Only if your DNS serves ONLY internal clients):

1. Open **Server Manager > Tools > DNS**
2. Right-click the DNS server (your server's hostname) and select **Properties**
3. Go to the **Advanced** tab
4. Check **Disable recursion (also disables forwarders)**
5. Click **OK**

Important: Only disable recursion on DNS servers that don't need to resolve external names. For domain controllers that need internet resolution, configure forwarders instead (see next section).

8. Configure DNS Forwarders Securely

Control how your DNS server resolves external queries.

Configure Standard Forwarders:

1. Open **Server Manager > Tools > DNS**
2. Right-click the DNS server (your server's hostname) and select **Properties**
3. Go to the **Forwarders** tab
4. Click **Edit**
5. Enter trusted DNS server IP addresses (examples below - use **YOUR preferred DNS**):
 - Google DNS: and
 - Cloudflare DNS: and
 - Or use your ISP's DNS servers
6. Click **OK**, then **OK** again

Configure Conditional Forwarders:

1. Open **Server Manager > Tools > DNS**
2. Expand the DNS server (your server's hostname)
3. Right-click **Conditional Forwarders** and select **New Conditional Forwarder**
4. Enter the domain name (e.g., - use the actual partner domain)

5. Enter the IP address(es) of that domain's DNS servers
 6. Check **Store this conditional forwarder in Active Directory, and replicate it as follows:**
 7. Select **All DNS servers in this domain**
 8. Click **OK**
-

9. Restrict DNS Zone Transfers

Prevent unauthorized servers from copying your DNS zone data.

Restrict Zone Transfers:

1. Open **Server Manager > Tools > DNS**
2. Expand **Forward Lookup Zones**
3. Right-click your zone (e.g., `domain.local`) - **use YOUR domain**) and select **Properties**
4. Go to the **Zone Transfers** tab
5. Check **Allow zone transfers**
6. Select **Only to the following servers**
7. Click **Edit**
8. Click **Add** and enter the IP address of each authorized secondary DNS server (e.g., `10.0.0.2`) - **use YOUR secondary DNS IPs**)
9. Click **OK**, then **OK** again

To Disable Zone Transfers Completely (if no secondary DNS servers):

1. Follow steps 1-4 above
 2. **Uncheck** "Allow zone transfers"
 3. Click **OK**
-

10. Enable DNS Socket Pool

Randomize source ports for DNS queries to prevent spoofing attacks.

Configure Socket Pool (Command Prompt - No GUI Available):

1. Open **Command Prompt as Administrator**
2. Run the following command:

```
cmd
```

```
dnscmd /Config /SocketPoolSize 10000
```

3. Verify the setting:

```
cmd
```

```
dnscmd /Info /SocketPoolSize
```

4. Restart DNS service:

```
cmd
```

```
net stop dns
```

```
net start dns
```

Or restart via Services:

1. Open **Server Manager > Tools > Services**
 2. Find **DNS Server**, right-click, select **Restart**
-

11. Configure DNS Firewall Rules

Restrict DNS traffic to authorized sources only.

Windows Firewall DNS Rules (Server Manager):

1. Open **Server Manager > Tools > Windows Defender Firewall with Advanced Security**
2. Click **Inbound Rules** in the left pane
3. Click **New Rule** in the right pane
4. Select **Port** and click **Next**
5. Select **TCP**, enter **53** in Specific local ports, click **Next**
6. Select **Allow the connection**, click **Next**
7. Check **Domain** profile only (uncheck Private and Public), click **Next**
8. Name: **DNS-TCP-Inbound-Internal**, click **Finish**
9. Right-click the new rule and select **Properties**

10. Go to **Scope** tab
11. Under **Remote IP address**, select **These IP addresses**
12. Click **Add** and enter your internal network ranges (e.g., 10.0.0.0/8) - use **YOUR network range**)
13. Click **OK**, then **OK** again
14. **Repeat steps 2-13 for UDP port 53** (name it **DNS-UDP-Inbound-Internal**)

Block External DNS (If applicable):

1. In **Windows Defender Firewall with Advanced Security**
 2. Click **Inbound Rules**, then **New Rule**
 3. Select **Port**, click **Next**
 4. Select **TCP and UDP**, enter **53**, click **Next**
 5. Select **Block the connection**, click **Next**
 6. Check **Public** profile only, click **Next**
 7. Name: **DNS-Block-External**, click **Finish**
-

12. Monitor DNS with Event IDs

Set up alerts for critical DNS events.

View DNS Events in Event Viewer:

1. Open **Server Manager > Tools > Event Viewer**
2. Expand **Applications and Services Logs > DNS Server**
3. Monitor the following Event IDs:

Event ID	Description	Action Required
150	DNS Server could not load zone from Active Directory	Check AD replication
408	DNS server could not bind to socket	Check for port conflicts
770	DNS server plugin DLL has caused an exception	Review DNS plugins
4000	DNS server zone loading errors	Check zone file integrity
4004	DNS server unable to create socket	Check network configuration
4013	DNS server zone transfer failed	Verify zone transfer settings
6004	DNS server encountered invalid data	Check for corrupted records
6527	DNS dynamic update failed	Check client permissions

Create Event Viewer Custom View for DNS Security:

1. In **Event Viewer**, right-click **Custom Views** and select **Create Custom View**
2. Select **By log**, check **DNS Server**
3. In **Event IDs**, enter:
4. Click **OK**
5. Name it **DNS Security Events**, click **OK**
6. You can now quickly view all critical DNS events in one place

13. Implement DNS Audit Logging via Group Policy

Track DNS configuration changes across all domain controllers.

Enable Audit Logging:

1. Open **Server Manager > Tools > Group Policy Management**
2. Expand **Forest > Domains > [YOUR DOMAIN] > Domain Controllers**
3. Right-click **Default Domain Controllers Policy** and select **Edit**
4. Navigate to: **Computer Configuration > Policies > Windows Settings > Security Settings > Advanced Audit Policy Configuration > Audit Policies > Object Access**
5. Double-click **Audit Directory Service Changes**

6. Check **Configure the following audit events**
 7. Check both **Success** and **Failure**
 8. Click **OK**
 9. Close Group Policy Management Editor
 10. Wait for Group Policy to update (or run `gpupdate /force` on the domain controller)
-

14. Regular DNS Security Maintenance

Weekly Checks (Manual):

1. Open **Server Manager > Tools > DNS**
2. Check for stale records:
 - Expand **Forward Lookup Zones**
 - Look for records with old timestamps
 - Right-click suspicious records > **Delete** if confirmed stale
3. Review DNS query logs:
 - Open `C:\Windows\System32\dns\dns.log` in Notepad
 - Look for unusual query patterns or high volumes from single IPs

Weekly Checks (PowerShell - Recommended):

```
powershell

# Check for stale DNS records (older than 30 days)
# Replace "domain.local" with YOUR domain name
Get-DnsServerResourceRecord -ZoneName "domain.local" |
  Where-Object {$_.TimeStamp -lt (Get-Date).AddDays(-30)} |
  Select HostName, RecordType, TimeStamp

# Review DNS query logs for anomalies
Get-WinEvent -LogName "DNS Server" -MaxEvents 1000 |
  Where-Object {$_.Id -in @(256,257,259,260)} |
  Group-Object Message | Sort Count -Descending
```

Monthly Checks (Event Viewer):

1. Open **Server Manager > Tools > Event Viewer**

2. Navigate to **Applications and Services Logs > DNS Server**
3. Look for Event IDs: **6001** (zone transfer started), **6002** (zone transfer completed)
4. Verify all zone transfers are legitimate

Monthly Checks (PowerShell):

```
powershell

# Audit DNS zone transfers from the last 30 days
Get-WinEvent -FilterHashtable @{
    LogName='DNS Server'
    ID=6001,6002
    StartTime=(Get-Date).AddDays(-30)
}

# Review DNS configuration changes from the last 30 days
Get-WinEvent -FilterHashtable @{
    LogName='Security'
    ID=5136,5137,5141
    StartTime=(Get-Date).AddDays(-30)
} | Where-Object {$_.Message -like "*DNS*"}
```

15. DNS Security Checklist

Use this checklist to verify your DNS security configuration:

- ☐ DNSSEC enabled and validated on all zones
- ☐ Dynamic updates set to "Secure only"
- ☐ DNS scavenging configured and enabled (7 day intervals)
- ☐ Cache locking set to 100% (PowerShell)
- ☐ Query logging enabled (Analytic logging recommended)
- ☐ Response Rate Limiting configured (Server 2016+)
- ☐ Recursion properly configured (disabled if internal-only)
- ☐ Forwarders configured to trusted DNS servers
- ☐ Zone transfers restricted to authorized servers only
- ☐ Socket pool enabled and set to 10000
- ☐ Firewall rules restrict DNS to internal networks only
- ☐ DNS security events monitored in Event Viewer
- ☐ Audit logging enabled via Group Policy

- ☐ Weekly and monthly maintenance tasks scheduled
 - ☐ Custom Event Viewer view created for DNS security
-

Quick Security Verification

Manual Verification Steps:

1. **Check DNSSEC:** Open DNS Manager > right-click your zone > **DNSSEC** > verify "Zone is signed"
2. **Check Dynamic Updates:** DNS Manager > right-click your zone > **Properties** > **General** tab > verify "Dynamic updates: Secure only"
3. **Check Scavenging:** DNS Manager > right-click DNS server > **Properties** > **Advanced** tab > verify "Enable automatic scavenging" is checked
4. **Check Zone Transfers:** DNS Manager > right-click your zone > **Properties** > **Zone Transfers** tab > verify restricted or disabled
5. **Check Event Logs:** Event Viewer > Custom Views > DNS Security Events (the one you created)

PowerShell Verification Script (Recommended):

```
powershell
```

```
# DNS Security Audit Script
# Replace "domain.local" with YOUR domain name throughout

Write-Host "=== DNS Security Audit ===" -ForegroundColor Cyan

# Check DNSSEC
$dnssec = Get-DnsServerDnsSecZoneSetting -ZoneName "domain.local" -ErrorAction SilentlyContinue
Write-Host "DNSSEC Enabled: $($dnssec.IsSignedWithKeyMaster)" -ForegroundColor $(if($dnssec.IsSignedWithKeyMaster){"Green"}else{"Red"})

# Check dynamic updates
$zone = Get-DnsServerZone -Name "domain.local"
Write-Host "Dynamic Updates: $($zone.DynamicUpdate)" -ForegroundColor $(if($zone.DynamicUpdate -eq "Secure"){"Green"}else{"Red"})

# Check scavenging
$scav = Get-DnsServerScavenging
Write-Host "Scavenging Enabled: $($scav.ScavengingState)" -ForegroundColor $(if($scav.ScavengingState){"Green"}else{"Red"})

# Check cache locking
$cache = Get-DnsServerCache
Write-Host "Cache Locking: $($cache.LockingPercent)%" -ForegroundColor $(if($cache.LockingPercent -eq 100){"Green"}else{"Red"})

# Check RRL (Server 2016+ only)
$rrl = Get-DnsServerResponseRateLimiting -ErrorAction SilentlyContinue
Write-Host "Response Rate Limiting: $($rrl.Mode)" -ForegroundColor $(if($rrl.Mode -eq "Enable"){"Green"}else{"Red"})

# Check zone transfers
Write-Host "Zone Transfer Settings: $($zone.SecureSecondaries)" -ForegroundColor $(if($zone.SecureSecondaries -ne "TransferFromMaster"){"Green"}else{"Red"})

Write-Host "`n=== Audit Complete ===" -ForegroundColor Cyan
```

Example Names and Placeholders Used in This Guide

Throughout this guide, you'll need to replace example names with your actual values:

Example Used	What to Replace With
<code>domain.local</code>	YOUR actual domain name (e.g., <code>contoso.com</code> , <code>company.local</code>)
<code>10.0.0.2</code> , <code>10.0.0.3</code>	YOUR actual secondary DNS server IP addresses
<code>10.0.0.0/8</code>	YOUR actual internal network IP range (e.g., <code>192.168.0.0/16</code>)
<code>partner.com</code>	Actual partner/external domain name you need conditional forwarding for
<code>8.8.8.8</code>	Your preferred external DNS forwarder (Google, Cloudflare, ISP, etc.)
"your server's hostname"	The name of YOUR domain controller as it appears in DNS Manager
"Domain Controllers OU"	This is standard and usually doesn't need changing
"Default Domain Controllers Policy"	This is standard and usually doesn't need changing

Additional Resources

- **Microsoft DNS Security Best Practices:** <https://docs.microsoft.com/windows-server/networking/dns/deploy/dns-security>
- **DNSSEC Deployment Guide:** <https://docs.microsoft.com/windows-server/networking/dns/deploy/dnssec>
- **DNS Logging and Diagnostics:** <https://docs.microsoft.com/troubleshoot/windows-server/networking/dns-logging-and-diagnostics>