DNS Open Resolver Mitigation Guide

Introduction

DNS open resolvers are systems that accept DNS queries from any source and can be abused for malicious activities like DDoS attacks. This guide provides step-by-step instructions to secure DNS servers across different operating systems, specifically tailored for Windows and Linux environments.

Steps to Follow if You Have an Open Resolver

If you discover that your DNS server is configured as an open resolver:

1. Immediately begin logging detailed DNS query data to identify potentially malicious activity.
2. Implement the below measures to restrict query sources and types.
3. Monitor network traffic for unusual patterns that could indicate an ongoing attack.

General Steps for Securing DNS Servers

1. **Identify Open Resolvers**:

- Start by checking whether your DNS servers are reachable from the internet. You can use tools like **dig** or online services use tools like DNSviz or DNSRecon to perform DNS queries against your servers from outside your network and identify any vulnerabilities.
- Analyze your DNS server configuration to ensure it's not configured as an open resolver. Open resolvers accept recursive DNS queries from any IP address, making them susceptible to abuse in DNS amplification attacks.

2. **Restrict Recursion**:

- Configure your DNS server software (e.g., BIND, Microsoft DNS Server) to only allow recursion from trusted clients. This prevents unauthorized users from exploiting your server for recursive queries.
- In BIND, you can achieve this by specifying the "**allow-recursion"** directive in your configuration file (**named.conf**). Only IP addresses listed in this directive will be allowed to perform recursive queries.

3. **Disable Recursion**:

- If your DNS server is purely authoritative and doesn't need to perform recursive queries for clients, consider disabling recursion altogether.
- Disabling recursion reduces the attack surface and minimizes the risk of abuse or exploitation.

4. **Implement Rate Limiting**:

- Configure your DNS server to limit the number of queries it accepts from a single IP address within a specified time frame. This helps mitigate the impact of DNS-based DDoS attacks and abusive behavior.
- Many DNS server software packages support built-in rate limiting mechanisms or can be supplemented with external tools or scripts.

5. **Use Firewalls**:

- Employ firewall rules to control DNS traffic to and from your DNS servers.
- Configure firewall rules to only permit DNS traffic from trusted sources, such as your internal network or specific IP addresses that require DNS services.
- Additionally, consider implementing DNS filtering at the firewall level to block known malicious domains and prevent communication with malicious DNS servers.

By following these steps and implementing the recommended measures, you can significantly enhance the security of your DNS servers and mitigate potential risks and vulnerabilities.

OS Specific Steps for Securing DNS Servers

1. Windows Server DNS Configuration

Objective:

Enhance security by preventing external DNS queries and mitigating the risk associated with open DNS resolvers.

Risk Overview:

Open DNS resolvers can be exploited for DNS amplification attacks, potentially involving your server in distributed denial-of-service (DDoS) attacks.

Configuration Strategy:

Utilize Windows Firewall to block external DNS queries, as Windows DNS Server does not support restricting recursive DNS requests at the server level.

Detailed Steps:

1. **Access Windows Firewall:**
    - Navigate to **Control Panel > System and Security > Windows Defender Firewall > Advanced Settings**.
2. **Configure Inbound Rules:**
    - Click on **Inbound Rules** in the left pane.
    - Select **New Rule** from the right pane.
    - Choose **Custom** and click **Next**.
    - Under Protocol and Ports, select **TCP** and specify port **53** (repeat for UDP if necessary).
    - Under Scope, add the IP addresses of your internal network in the **remote IP addresses** section.
3. **Implement Rule:**
    - Set the action to **Allow the connection**.
    - Apply the rule only to the domain and private network profiles to avoid blocking legitimate external traffic necessary for other services.
    - Name the rule descriptively, e.g., "Restrict DNS Queries to Internal Network."

Validation:

Monitoring and Logging:

- Enable logging in Windows Firewall for the created rule to monitor any attempted access.
- Regularly review the firewall logs through the **Windows Event Viewer** to ensure compliance and detect anomalies.

Windows Server - Option 2: Disable DNS Recursion

Detailed Steps:

1. **Access DNS Manager:**
   - Navigate to **Control Panel > Administrative Tools > DNS** to open DNS Manager.
2. **Modify Server Properties:**
   - Right-click your DNS server from the list and select **Properties.**
3. **Configure Advanced Settings:**
   - Go to the **Advanced** tab.
   - Check the box labelled **Disable recursion** (also disables forwarders).
4. **Apply Changes:**
   - Click **Apply**, then **OK** to save the settings.

Validation:

- **Monitoring and Logging:**
  - Ensure logging is enabled for DNS queries to track the effects of disabling recursion.
  - Use the DNS Manager and Windows Event Viewer to review logs and verify that no recursive queries are being processed.

2. Linux/BIND Server Configuration

Objective:

Restrict DNS recursion exclusively to local networks to secure BIND DNS servers.

Risk Overview:

Unsecured BIND servers are vulnerable to being used for DNS amplification attacks due to their recursive query capabilities.

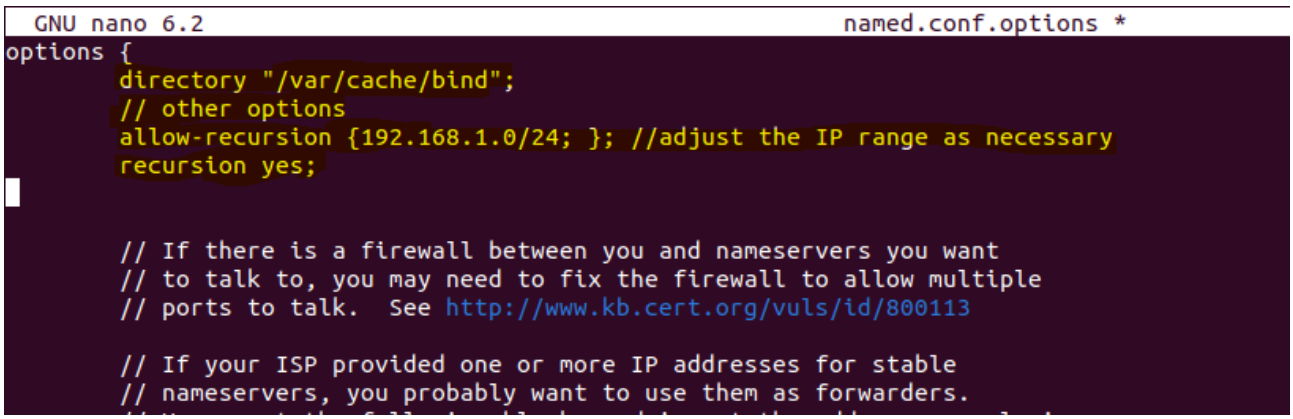
Configuration Strategy:

Utilize the "**allow-recursion"** directive in BIND to restrict recursive queries to specified IP ranges.

Detailed Steps:

1. Configure **named.conf:**

- Edit the BIND configuration file, typically located at **"/etc/bind/named.conf".**
- Insert or modify the allow-recursion directive within the options block as follows:



- Save the changes and restart the BIND service to apply the configuration:

  sudo systemctl restart bind9

- Ensure no errors are reported during restart.

Validation:

- Use DNS query tools such as `dig` from an external network to verify that recursion is now restricted.

Monitoring:

- Regularly inspect **/var/log/named/bind.log** for unauthorized access attempts or other anomalies.

3. MacOS DNS Configuration Best Practices

Detailed Steps:

1. **Access Network Settings:**
   - Navigate to **System Preferences > Network.** Select the network interface you are using (e.g., Ethernet, Wi-Fi) and click **Advanced.**
2. **Configure DNS Settings:**
   - Go to the **DNS** tab.
   - Click the + button under DNS Servers to add DNS servers that you trust or are within your internal network.
   - Remove any unknown or untrusted DNS servers from the list to prevent DNS hijacking.
3. **Apply and Lock DNS Settings:**
   - Click **OK** to save the DNS settings.
   - Back in the main Network window, click **Apply** to enforce the new settings.
4. **Enhance Security with DNS over HTTPS (DoH) or DNS over TLS (DoT):**
   - MacOS does not natively support DoH or DoT through the system settings as of my last update, but these can be configured through third-party applications like DNSCrypt or by using browsers that support these protocols, such as Firefox or Chrome.

Validation:

- **Regular Updates:**
  - Keep your MacOS and all network applications updated to ensure you have the latest security patches.
- **Monitoring Network Activity:**
  - Use tools such as Little Snitch or the built-in Terminal to monitor outbound connections, including DNS requests, to identify and block unwanted or malicious traffic.

4. For Cisco Devices

Detailed Steps:

1. **Access Router Configuration:**

   Log into your Cisco router or firewall using SSH or a console port.

2. **Enter Configuration Mode:**

   Type **configure terminal** to enter the global configuration mode.

3. **Define Access Control Lists (ACL):**

   Create an ACL to define which IPs are allowed to make DNS requests:

   ```
   ip access-list extended ALLOW_DNS
   permit udp any host [Your_Server_IP] eq 53
   deny udp any any eq 53
   permit ip any any
   ```

   Replace［**Your_Server_IP**］with the IP address of your DNS server.
4. **Apply ACL to the Incoming Interface:**

   Apply the ACL to the interface receiving external traffic:

   ```
   interface [Your_Interface]
   ip access-group ALLOW_DNS in
   ```

   Replace［**Your_Interface**］with the interface connected to the external network.

5. **Save Configuration:**

   Save your changes with **write memory.**

Validation:

- **Test Configuration:**
  - Test DNS functionality internally and ensure external DNS queries are blocked.
- **Monitoring and Logging:**
  - Enable logging on the ACL to monitor denied requests and ensure compliance.

5. For Fortinet Devices

Detailed Steps:

1. **Access FortiGate Interface:**
   - Log into the FortiGate firewall using the web interface or FortiClient.
2. **Create IPv4 Policy:**
   - Go to **Policy & Objects > IPv4 Policy** and create a new policy.
   - Set incoming interface to your external facing interface.
   - Set outgoing interface to the internal network facing the DNS server.
3. **Configure Policy Settings:**
   - Under **Source**, add the external IP ranges to be blocked or allowed.
   - Under **Destination**, specify your DNS server's IP and set service to `DNS`.
   - Action should be set to **DENY** for unwanted sources and **ACCEPT** for trusted sources.
4. **Enable Logging:**
   - Enable logging to record all sessions to track allowed and blocked requests.
5. **Apply and Save:**
   - Apply the changes and save the configuration.

Validation:

- **Testing:**
  - Perform tests from external and internal networks to validate that only allowed IPs can query your DNS.
- **Review Logs:**
  - Regularly check the firewall logs to ensure that the rules are enforced correctly.

General Security Measures for All Systems

1. **Regular Updates:**
   - Maintain all systems with the latest security patches and updates to mitigate vulnerabilities.
2. **Disable Unnecessary Services:**
   - Deactivate any services not required on the DNS server to minimize the attack surface.
3. **Conduct Regular Security Audits:**
   - Periodically perform security audits and vulnerability scans to identify and address security gaps.
4. **Educate and Train Staff:**
   - Provide ongoing training for staff to recognize security risks and follow best practices.

Conclusion

By following the detailed steps outlined above for each system type, administrators can secure their DNS servers against misuse as open resolvers.