

ATTACKING PFSENSE

**FIREWALL/ROUTER
SECURE DEV**



Attacking pfSense

• Aug 19, 2024 •  27 min read

Table of contents

AutoConfigBackup

Profile Applicability

Description

Rationale

Attack Scenario

Audit Procedure

Remediation Procedure

Default Value

Example Commands (if available for CLI)

Message Of The Day (MOTD)

Rationale

- › Attack Scenario
- › Audit Procedure
- › Remediation Procedure
- › Default Value

Hostname

- › Rationale
- › Attack Scenario
- › Audit Procedure

- › Remediation Procedure
- › Default Value

DNS server

- › Rationale
- › Attack Scenario
- › Audit Procedure
- › Remediation Procedure
- › Default Value
- › Example Command Sequence

DNS Rebinding

- › Rationale
- › Attack Scenario
- › Audit Procedure
- › Remediation Procedure
- › Default Value
- › Example Command Sequence

IPv6

- › Rationale
- › Attack Scenario
- › Audit Procedure
- › Default Value

HTTPS :)

- › Rationale
- › Attack Scenario
- › Audit Procedure
- › Remediation Procedure

- › Default Value

LDAP or RADIUS as promise

- › Rationale
- › Attack Scenario
- › Audit Procedure
- › Remediation Procedure
- › Default Value

Console Menu

- › Rationale:
- › Attack Scenario:
- › Audit Procedure:
- › Remediation Procedure:
- › Default Value:

Ensure All Default Accounts are Either Disabled or Utilize Strong Passwords (Manual)

- › Profile Applicability:
- › Description:
- › Rationale:
- › Attack Scenario:
- › Audit Procedure:
- › Remediation Procedure:
- › Default Value:

Local Account Status

- › Rationale:
- › Attack Scenario:
- › Audit Procedure:
- › Remediation Procedure:

- › Default Values:
- › Combined Security Impact:

ICMP Request

- › Rationale:
- › Attack Scenario:
- › Audit Procedure:
- › Remediation Procedure:
- › Default Configuration:

SNMP

- › Rationale:
- › Attack Scenario:
- › Audit Procedure:
- › Remediation Procedure:
- › Default Values:

Resources

Show less ^

Attacking pfSense, a popular open-source firewall and router platform, typically involves targeting its various components and configurations to exploit vulnerabilities and gain unauthorized access. Given its widespread use in securing networks, attackers often focus on exploiting misconfigurations or weaknesses in pfSense setups, such as inadequate firewall rules, unpatched vulnerabilities, or poorly configured services. Common attack vectors include exploiting default settings, misconfigured services like SNMP or DNS, and weaknesses in authentication mechanisms. These vulnerabilities can lead to unauthorized access, data breaches, and disruptions in network security, making it crucial for administrators to maintain stringent security practices.

In addition to exploiting configuration flaws, attackers might leverage sophisticated techniques such as DNS rebinding or brute-force attacks to compromise pfSense. DNS rebinding can be used to trick the firewall into exposing internal resources, while brute-force attempts target weak passwords or inadequate login protection settings. Ensuring robust security measures, such as disabling unused services, enforcing strong passwords, and configuring secure protocols, is essential to protect pfSense from these threats. Regular security audits and adherence to best practices are vital for maintaining the integrity and security of pfSense deployments against evolving attack techniques.

AutoConfigBackup

To ensure that the `AutoConfigBackup` feature is enabled, you can follow the steps outlined below. This is essential for maintaining a reliable backup process, which can be critical in the event of a configuration error or unintended change.

Profile Applicability

- **Level 1**

Description

Enabling AutoConfigBackup ensures that a backup of the system configuration is automatically created before and after significant changes, as well as periodically. This is crucial for quick recovery if a change leads to unfavorable results. It is a best practice to maintain these backups as part of a robust configuration management strategy.

Rationale

Having an automatic backup system helps maintain a safety net for configuration management. If a change results in unexpected issues, the backup can be used to restore the system to its previous state. Even in scenarios where manual backups are

overlooked, AutoConfigBackup ensures that recent configurations are saved, minimizing the risk of losing important data.

Attack Scenario

An attacker could exploit the absence of an AutoConfigBackup to modify system configurations and potentially disrupt services or introduce vulnerabilities. Without automatic backups, restoring the system to a stable state would be more difficult, giving the attacker more time to cause damage or remain undetected.

For instance, an attacker might gain unauthorized access to the system and alter critical configuration settings. If AutoConfigBackup is disabled, the system would have no recent backups to revert to, making it harder to recover from the attack quickly. On the other hand, if AutoConfigBackup is enabled, the system administrator can promptly restore the last known good configuration, mitigating the attack's impact.

Audit Procedure

To verify that AutoConfigBackup is enabled:

1. In the GUI:

- Navigate to `Services > Auto Config Backup`.
- Check if the AutoConfigBackup service is enabled.

Remediation Procedure

To enable AutoConfigBackup:

1. In the GUI:

- Navigate to `Services > Auto Config Backup`.
- Click on the `Settings` at the top of the page.

- Check the option `Enable ACB`.
- Click `Save` to apply the changes.

Default Value

- `Disabled`

Example Commands (if available for CLI)

If your environment allows command-line configuration, you might use the following steps to check and enable `AutoConfigBackup` via CLI.

```
# Check if AutoConfigBackup is enabled
show service auto-config-backup

# Enable AutoConfigBackup (pseudo-command)
set service auto-config-backup enable
commit
save
```

COPY 

Message Of The Day (MOTD)

The Message of the Day (MOTD) is a text message displayed to users upon login to a system. Setting the MOTD on your system provides an important security measure by presenting legal disclaimers, notices, or other critical information to users. This is a crucial step in reinforcing security policies and ensuring that users are aware of their rights, responsibilities, and the legal implications of using the system.

Rationale

MOTD banners serve several legal and security purposes:

1. **Consent to Monitoring:** By displaying a clear warning that the system is subject to monitoring, users implicitly consent to such monitoring, which is important for compliance with laws like Title III.
2. **Consent to Data Retrieval:** The banner can also inform users that their stored files and records may be accessed, helping comply with the Electronic Communications Privacy Act (ECPA).
3. **No Expectation of Privacy:** For government systems, the MOTD can help eliminate any expectation of privacy by notifying users that their activity is not private, in line with legal precedents like *O'Connor v. Ortega*.
4. **General Awareness:** It informs users of important security policies and terms of use, which can help mitigate unauthorized actions or misunderstandings about system usage.

Attack Scenario

An attacker may try to exploit the absence of a proper MOTD by gaining unauthorized access to the system and claiming ignorance of the system's monitoring or legal requirements. For example, without an MOTD banner, an attacker could argue that they were unaware that their actions were being logged or that they had no warning of the system's monitoring policies.

Moreover, if a malicious insider accesses sensitive data or misuses the system, they could attempt to claim a "reasonable expectation of privacy" if no MOTD is in place to counter that expectation. In contrast, a clearly stated MOTD helps to mitigate these risks by providing unambiguous notice of monitoring and legal policies.

Audit Procedure

To verify that the MOTD is set correctly:

1. **In pfSense Shell:**

Open the terminal and type the following command to display the current MOTD:

```
cat /etc/motd
```

1. • Review the content to ensure it includes the appropriate legal and security notices.

Remediation Procedure

To set or modify the MOTD:

1. In pfSense Shell:

Open the terminal and use a text editor (such as `vi`) to edit the MOTD file:

```
vi /etc/motd
```

- Enter the desired MOTD message. This message should include necessary legal disclaimers, consent to monitoring, and any other relevant information.
- Save the changes and exit the editor.

For example, you might include the following MOTD message:

```
*****  
WARNING: Unauthorized access to this system is prohibited.  
All activities on this system are logged and monitored.  
By using this system, you consent to such monitoring and logging.  
*****
```

Default Value

The default MOTD message in pfSense typically includes general information about the FreeBSD operating system, with no specific security or legal notices.

COPY 

```
[2.5.0-RELEASE][admin@pfSense.home.arpa]/root: cat /etc/motd
FreeBSD ?.?.? (UNKNOWN)
Welcome to FreeBSD!
Release Notes, Errata: https://www.FreeBSD.org/releases/
Security Advisories: https://www.FreeBSD.org/security/
FreeBSD Handbook: https://www.FreeBSD.org/handbook/
FreeBSD FAQ: https://www.FreeBSD.org/faq/
Questions List: https://lists.FreeBSD.org/mailman/listinfo/freebsd-questions
FreeBSD Forums: https://forums.FreeBSD.org/
Documents installed with the system are in the /usr/local/share/doc/freebsd
directory, or can be installed later with: pkg install en-freebsd-doc
For other languages, replace "en" with a language code like de or fr.
Show the version of FreeBSD installed: freebsd-version ; uname -a
Please include that output and any error messages when posting questions.
Introduction to manual pages: man man
FreeBSD directory layout: man hier
Edit /etc/motd to change this login announcement.
```

Hostname

Setting a unique and descriptive hostname for your device is important for asset management, network organization, and security. The hostname is a label assigned to a device on a network, and it is used to identify the device in various administrative and operational tasks.

Rationale

A properly set hostname is vital for several reasons:

1. **Asset Management:** It helps in the proper identification of devices across the network, which is crucial for maintaining an accurate inventory.
2. **Security:** A well-defined hostname assists in deploying public keys, certificates, and managing access controls. It also plays a significant role in correlating logs from different systems, which is critical during incident response.
3. **Operational Efficiency:** When managing a large network, descriptive hostnames make it easier to identify and troubleshoot devices, avoiding confusion that could arise from using default or non-descriptive names.

Attack Scenario

An attacker could exploit a default or poorly configured hostname by blending in with legitimate network traffic, making it harder for administrators to detect unauthorized devices. For instance, if all devices use the default hostname (e.g., "pfSense"), it becomes challenging to distinguish between authorized and unauthorized devices.

Additionally, during a security incident, if the hostnames are not uniquely set, correlating logs and identifying affected devices could be significantly delayed, providing attackers with more time to exploit the network.

Audit Procedure

To verify the current hostname:

1. **In the CLI:**
 - Open the terminal and type the following command

```
hostname
```

COPY 

1.
 - The current hostname of the device will be displayed.
2. **In the GUI:**

- Navigate to `System > General Setup`.
- Check the `Hostname` field to see if it is set to a unique and descriptive name.

Remediation Procedure

To set or change the hostname:

1. In the GUI:

- Navigate to `System > General Setup`.
 - Locate the `Hostname` field.
 - Enter a new, unique hostname that clearly identifies the device.
 - Click `Save` to apply the changes
- Example:
 - If the device is a firewall at a branch office in New York, you might set the hostname to `nyc-branch-fw`.
 - **In the CLI** (Optional, depending on the system):
 - You can temporarily change the hostname using the following command:

COPY 

```
hostname new-hostname
```

1.
 - To make the change permanent, you might need to edit the system's hostname configuration file (e.g., `/etc/hostname` on some Unix systems), though the GUI method is typically preferred in systems like pfSense.

Default Value

- The default hostname for a pfSense device is typically `pfSense`.

DNS server

Configuring a DNS (Domain Name System) server on your system is critical for resolving domain names into IP addresses, allowing proper network communication. A primary DNS server should be specified for the system, and secondary or tertiary DNS servers can be added as backups. These DNS servers ensure that hostname resolution occurs smoothly and efficiently across the network.

Rationale

DNS configuration is crucial for several reasons:

1. **Hostname Resolution:** DNS servers translate human-readable domain names (e.g., `example.com`) into IP addresses that computers use to identify each other on the network.
2. **Redundancy:** Configuring multiple DNS servers ensures that if one server fails or times out, the system can still resolve domain names using the backup servers.
3. **Security:** Using secure and trusted DNS servers can help prevent DNS spoofing or other DNS-related attacks. It also ensures that the system can reliably connect to trusted hosts and services.

Attack Scenario

An attacker could exploit a misconfigured or absent DNS server setting to redirect traffic to malicious sites. For instance, if the DNS server is not configured or is set to a rogue DNS server, an attacker could perform a man-in-the-middle attack, redirecting users to phishing sites or intercepting sensitive data.

In another scenario, if a DNS server fails and there is no secondary DNS server configured, the system may be unable to resolve domain names, resulting in a denial of service. This could be particularly disruptive in environments where continuous network access is critical.

Audit Procedure

To verify that a DNS server is properly configured:

1. In the GUI:

- Navigate to `System > General Setup`.
- Look at the 'DNS Servers' field in the 'DNS Server Settings' table.
- Ensure that at least one DNS server is listed, and preferably, that secondary and tertiary servers are also configured.

Remediation Procedure

To configure or update the DNS server settings:

1. In the GUI:

- Navigate to `System > General Setup`.
- In the 'DNS Server Settings' table, enter the IP addresses of your primary, secondary, and tertiary DNS servers in the 'DNS Servers' fields.
- Click `Save` to apply the changes.

Example:

- Primary DNS Server: `8.8.8.8` (Google DNS)
- Secondary DNS Server: `8.8.4.4` (Google DNS)
- Tertiary DNS Server: `1.1.1.1` (Cloudflare DNS)

2. Optional CLI Check:

- To check the current DNS server configuration via the command line, you can use:

```
cat /etc/resolv.conf
```

- This file typically shows the DNS servers that the system is currently using. Make sure it lists the correct DNS server IP addresses.

Default Value

- By default, the DNS server field may be blank unless the WAN type is set to DHCP, in which case the ISP may assign DNS servers automatically.

Example Command Sequence

To summarize the process:

1. Audit:

- GUI: System > General Setup > DNS Servers

2. CLI (Optional):

```
cat /etc/resolv.conf
```

COPY 

Remediation:

- GUI: System > General Setup > DNS Servers , enter the desired DNS servers, and save.

DNS Rebinding

The "DNS Rebind Check" feature in pfSense helps protect against DNS rebinding attacks. These attacks involve manipulating DNS responses to direct internal, private IP addresses to external attackers, potentially exposing internal network resources. While this feature is generally important for security, certain configurations may require it to be disabled.

Rationale

DNS rebinding attacks allow an attacker to bypass the browser's same-origin policy, enabling them to gain unauthorized access to internal networks and systems. By exploiting a DNS server's ability to resolve domain names to private IP addresses,

attackers can execute malicious scripts or access sensitive information that would otherwise be protected.

The "DNS Rebind Check" feature in pfSense prevents this by blocking DNS responses that resolve domain names to private IP addresses, which are typically used within internal networks. Disabling this feature, as advised in some specific circumstances, would expose the system to such attacks, which is why it's generally recommended to leave it enabled.

Attack Scenario

If the "DNS Rebind Check" is unchecked (disabled), an attacker could leverage DNS rebinding techniques to gain access to internal resources that should not be exposed externally. For example, an attacker might register a domain and configure its DNS records to point to a private IP address. When a user within the network accesses this domain, the attacker's script could interact with internal systems using the victim's browser as a proxy, potentially leading to data exfiltration or other malicious actions.

Audit Procedure

To verify if the "DNS Rebind Check" is enabled:

1. In the GUI:

- Navigate to `System > Advanced`.
- Look for the "DNS Rebind Check" field.
- Ensure that the checkbox for "DNS Rebind Check" is checked.

Remediation Procedure

To disable the "DNS Rebind Check" (if required for a specific configuration):

1. In the GUI:

- Navigate to `System > Advanced`.
- Find the "DNS Rebind Check" field.

- Uncheck the "DNS Rebind Check" checkbox.
- Click **Save** to apply the changes.

Note: Disabling this feature increases the risk of DNS rebinding attacks and should only be done if absolutely necessary and with other compensating controls in place.

Default Value

- By default, the "DNS Rebind Check" is **enabled** (i.e., the checkbox is checked).

Example Command Sequence

To summarize the process:

1. Audit:

- GUI: **System > Advanced > DNS Rebind Check**, verify that the checkbox is checked.

2. Remediation:

- GUI: **System > Advanced > DNS Rebind Check**, uncheck the checkbox (if necessary), and save the settings.

IPv6

IPv6 is the next generation of the Internet Protocol (IP), designed to replace IPv4 due to the latter's limitations, particularly in address space. However, not all organizations have adopted IPv6 or dual-stack configurations (which run both IPv4 and IPv6). If IPv6 is not in use within your network, it is recommended to disable it to reduce the system's attack surface.

Rationale

Disabling IPv6 when it is not in use can reduce potential vulnerabilities associated with its presence on a network. IPv6 introduces new features and complexities, which, if not properly managed or needed, can provide additional avenues for

attackers to exploit. Disabling IPv6 helps to mitigate these risks by ensuring that only the necessary network protocols are active, thereby minimizing the attack surface.

Attack Scenario

If IPv6 is enabled but not actively used or monitored, it could expose the network to attacks. For example, an attacker could exploit IPv6-related vulnerabilities or misconfigurations to gain unauthorized access to network resources. If the network administrators are not actively monitoring IPv6 traffic (because they believe the network is IPv4-only), this traffic could be used to bypass security controls, perform reconnaissance, or establish covert communication channels within the network.

Audit Procedure

To verify that IPv6 is disabled:

1. In the CLI:

Check the system configuration file for IPv6 settings:

```
cat /etc/sysctl.conf
```

COPY 

Look for the following entries

```
net.ipv6.conf.all.disable_ipv6=1  
net.ipv6.conf.default.disable_ipv6=1  
net.ipv6.conf.lo.disable_ipv6=1
```

COPY 

1. In the GUI:

- Navigate to **System > Advanced > Networking**.

- Check if the "Allow IPv6" option is unchecked. If it is checked, IPv6 is enabled.

Default Value

- By default, the "Allow IPv6" option in the GUI is checked, meaning IPv6 is enabled unless explicitly disabled.

HTTPS :)

The Web Management interface (Web Admin Management Portal) of network devices, such as pfSense, should be accessed using the HTTPS protocol instead of HTTP. HTTPS encrypts the data transmitted between the web browser and the management interface, protecting sensitive information such as login credentials from being intercepted by attackers.

Rationale

Using HTTP for the Web Management interface is inherently insecure as it transmits all data, including sensitive information like passwords, in plain text over the network. This lack of encryption makes it easy for attackers to intercept and view the data using tools like packet sniffers. By configuring the Web Management Portal to use HTTPS, all data is encrypted, ensuring confidentiality and integrity while also providing assurance about the identity of the connected parties. HTTPS also prevents man-in-the-middle attacks, where an attacker could intercept and alter communications between the user and the management interface.

Attack Scenario

If the Web Management Portal is accessible via HTTP, an attacker on the same network could intercept the unencrypted traffic using a packet-sniffing tool (e.g., Wireshark). By doing so, the attacker could capture sensitive data, including usernames, passwords, session cookies, or any other information exchanged during the session. This captured data could then be used to gain unauthorized access to

the network device, potentially leading to further attacks, such as configuration changes, data breaches, or denial-of-service attacks.

In another scenario, an attacker could perform a man-in-the-middle (MITM) attack, intercepting and modifying the HTTP traffic to inject malicious commands or capture sensitive information without the user's knowledge.

Audit Procedure

To verify if HTTPS is enabled for Web Management:

1. In the GUI:

- Navigate to `System > Advanced > Admin Access`.
- Check the `webConfigurator` section.
- Ensure that HTTPS is selected as the protocol.

Remediation Procedure

To configure the Web Management interface to use HTTPS:

1. In the GUI:

- Navigate to `System > Advanced > Admin Access`.
- Locate the `webConfigurator` section.
- In the `Protocol` dropdown menu, select `HTTPS` (SSL/TLS).
- Click `Save` to apply the changes.

Note: Before switching to HTTPS, ensure that a valid SSL/TLS certificate is configured. You can either generate a self-signed certificate or use a certificate issued by a trusted Certificate Authority (CA).

Default Value

- By default, the **Protocol** in the **System > Advanced > Admin Access** section is set to **HTTPS (SSL/TLS)**.

LDAP or RADIUS as promise

Configuring an LDAP (Lightweight Directory Access Protocol) or RADIUS (Remote Authentication Dial-In User Service) server provides a centralized authentication system for managing user access to network devices. Centralized authentication allows for more streamlined user management, enhanced security, and the ability to implement consistent access controls across multiple devices.

Rationale

Implementing a centralized authentication, authorization, and accounting (AAA) system using LDAP or RADIUS servers offers several security benefits:

1. **Centralized Management:** It provides a single point for managing user credentials and access policies, simplifying administration.
2. **Enhanced Security:** By centralizing authentication, organizations can enforce stronger password policies, multi-factor authentication (MFA), and other security measures more effectively.
3. **Auditing and Accountability:** RADIUS and LDAP servers offer detailed logging and monitoring capabilities, allowing for better tracking of user activity and access attempts, aiding in incident response and compliance.

Without centralized authentication, user credentials are often stored locally on each device, which can lead to inconsistent access controls, increased administrative overhead, and greater vulnerability to security breaches.

Attack Scenario

If LDAP or RADIUS is not configured and local authentication is used instead, the system is at higher risk for several types of attacks:

1. **Credential Harvesting:** If an attacker compromises the local authentication database on one device, they could potentially gain access to all user credentials stored on that device.
2. **Inconsistent Policies:** Without centralized management, enforcing consistent password policies, access control, and user deprovisioning across multiple devices becomes difficult, potentially leading to weaker security on some devices.
3. **Brute Force Attacks:** Attackers might attempt brute-force attacks against local accounts, especially if password policies are not uniformly enforced.

In contrast, a centrally managed LDAP or RADIUS server can enforce strong, consistent authentication policies, reducing these risks.

Audit Procedure

To verify if an LDAP or RADIUS server is configured:

1. In the GUI:

- Navigate to `System > User Manager`.
- Click the `Authentication Servers` tab at the top.
- Check the "Authentication Servers" setting to see if an LDAP or RADIUS server is configured.

Remediation Procedure

To configure an LDAP or RADIUS server for authentication:

1. In the GUI:

- Navigate to `System > User Manager`.
- Click the `Authentication Servers` tab at the top.
- Click `+ Add` to configure a new LDAP or RADIUS server.

- Fill in the necessary details for the LDAP or RADIUS server, including:
 - **Server Name:** A descriptive name for the server.
 - **Type:** Select either `LDAP` or `RADIUS`.
 - **Hostname or IP Address:** The address of the LDAP or RADIUS server.
 - **Port:** The port number used by the server (default is usually 389 for LDAP and 1812 for RADIUS).
 - **Base DN (for LDAP):** The distinguished name (DN) from which to start searching for user accounts.
 - **RADIUS Shared Secret (for RADIUS):** A shared secret key used for communication with the RADIUS server.
- Click `Save` to apply the configuration.

Default Value

- By default, the `Server Name` in the `System > User Manager > Authentication Servers` setting is set to `Local Database`, meaning local authentication is used.

Console Menu

This control ensures that the Console Menu of the firewall is password-protected. If the Console Menu is not protected by a password, anyone with physical or remote access to the console could potentially alter the configuration of the firewall or gain unauthorized access to the network.

Rationale:

Leaving the Console Menu unprotected allows unauthorized individuals to access and potentially compromise the firewall. For example, if someone gains physical access to the device or remote access to the console, they could modify the firewall's settings, disable security features, or even lock out legitimate administrators. Protecting the

Console Menu with a password mitigates this risk by ensuring that only authorized users can access the management interface.

Attack Scenario:

Consider a scenario where an attacker gains physical access to a network device with an unprotected Console Menu. The attacker could connect to the device and gain full access to the firewall's configuration settings, potentially disabling firewall rules, changing network configurations, or installing malicious firmware. This could result in a significant security breach, compromising the entire network. Additionally, if the attacker is skilled enough, they could create a backdoor that would allow future access without detection.

Audit Procedure:

To verify if the Console Menu is password-protected:

1. In the GUI:

- Navigate to `System > Advanced > Admin Access`.
- Check the "Console Options" setting to ensure that the Console Menu is set to require a password.

Remediation Procedure:

To configure the Console Menu to require a password:

1. In the GUI:

- Navigate to `System > Advanced > Admin Access`.
- Locate the "Console Options" section.
- Enable the setting to password-protect the Console Menu.
- Click "Save" to apply the changes.

Default Value:

- By default, the "Console menu" option is unchecked, meaning it is not password-protected.

Ensure All Default Accounts are Either Disabled or Utilize Strong Passwords (Manual)

Profile Applicability:

- Level 1

Description:

Ensure that all default accounts on the system are either disabled or have strong, unique passwords. This includes well-known default accounts such as `admin`, `guest`, `user`, `root`, `administrator`, `operator`, `supervisor`, and `demo`.

Rationale:

Default accounts are often targeted by attackers because their credentials are widely known. If these accounts are left enabled with their default passwords, they provide an easy entry point for attackers. Disabling unnecessary default accounts and securing necessary ones with strong passwords significantly reduces the risk of unauthorized access.

Attack Scenario:

An attacker who knows the default credentials for a system (e.g., `admin/admin`) can easily gain unauthorized access if these accounts have not been disabled or secured with strong passwords. Once inside, the attacker could escalate privileges, alter system configurations, extract sensitive data, or pivot to other systems within the network. For example, if a default `admin` account with a weak password is left enabled, an attacker could use automated tools to brute-force the password and gain full administrative control over the system.

Audit Procedure:

To verify the status of default accounts:

1. In the GUI:

- Navigate to `System > User Manager > Users` .
- Review the list of users to identify any default accounts that are still active and ensure they are either disabled or have strong passwords.

Remediation Procedure:

To disable or secure default accounts:

1. In the GUI:

- Navigate to `System > User Manager > Users` .
- Identify any default users that are not necessary.
- Disable or delete any unnecessary default accounts.
- For necessary accounts (like `admin`), ensure they are secured with strong, unique passwords.
- Note: The default `admin` account should remain enabled for high availability synchronization, but it must be secured with a strong password to prevent unauthorized access.

Default Value:

- The default user is `admin` .

Local Account Status

This combined control focuses on disabling non-admin local user accounts and configuring secure login protection settings to mitigate unauthorized access risks. Disabling non-admin local accounts prevents the use of unmanaged, potentially insecure credentials, while setting proper login protection thresholds and lockout times protects against brute-force attacks.

Rationale:

Local accounts can pose a security risk because they bypass centralized management systems like LDAP or RADIUS, making them difficult to monitor and manage. Disabling all local accounts except for the necessary admin user reduces the attack surface. Additionally, configuring secure login protection settings ensures that brute-force attacks are thwarted by limiting login attempts and enforcing account lockouts after failed attempts.

Attack Scenario:

1. Local Account Exploitation:

- If local user accounts are left enabled, an attacker with knowledge of a default or weak password could gain unauthorized access. For example, if a non-admin account is compromised, the attacker could escalate privileges or pivot to other systems, leading to a full network compromise.

2. Brute-Force Attack:

- An attacker could attempt to brute-force the login page by trying numerous password combinations. Without a properly configured login protection threshold, the attacker might eventually succeed. However, with a threshold set to 30 or less and a lockout time of 300 seconds or more, the system will block further attempts, making it difficult for the attacker to break in.

Audit Procedure:

1. Check Local Account Status:

- **Manual Audit:** Review each user account to ensure that all non-admin local accounts are disabled.

2. Audit Login Protection Threshold:

- **In the GUI:**
 - Navigate to `System > Advanced`.

- Check the **Login Protection** settings, specifically the **Threshold** field, to ensure it is set to 30 or less.

3. Audit Access Block Time:

- **In the GUI:**
 - Navigate to **System > Advanced > Admin Access**.
 - Verify that the **Blocktime** value in the **Login Protection** settings is set to 300 seconds or more.

Remediation Procedure:

1. Disable Non-Admin Local Accounts:

- **Manual Process:**
 - Navigate to **System > User Manager > Users**.
 - Review all user accounts and disable any local accounts that are not necessary, keeping only the **admin** account enabled.

2. Set Login Protection Threshold:

- **In the GUI:**
 - Navigate to **System > Advanced**.
 - Set the **Login Protection** threshold to 30 or less in the **Threshold** field.
 - Click **Save**.

3. Set Access Block Time:

- **In the GUI:**
 - Navigate to **System > Advanced > Admin Access**.
 - Set the **Blocktime** value to 300 seconds or more in the **Login Protection** settings.
 - Click **Save**.

Default Values:

- **Local Accounts:** Only the `admin` user is enabled by default.
- **Login Protection Threshold:** Default value is 30.
- **Blocktime:** Default is typically blank, requiring manual configuration.

Combined Security Impact:

By disabling unnecessary local accounts and configuring robust login protection measures, you significantly reduce the risk of unauthorized access through compromised accounts or brute-force attacks. These settings collectively harden the system against common attack vectors, enhancing the overall security posture of the network infrastructure. Regular audits and updates to these settings are essential to maintaining effective defense mechanisms against evolving threats.

ICMP Request

ICMP (Internet Control Message Protocol) facilitates network communication and troubleshooting by sending error messages and operational information. However, allowing unrestricted ICMP traffic can expose the network to various attacks, such as Ping Floods or ICMP Redirects. It is crucial to configure ICMP rules on the firewall to limit the types of ICMP requests and ensure that only necessary types are permitted.

Rationale:

While ICMP is valuable for network diagnostics and management, it can also be exploited by attackers. For example, a Ping Flood attack can overwhelm a network with excessive ICMP Echo Requests, leading to denial of service. Similarly, ICMP Redirects can be used to manipulate routing tables maliciously. By securely configuring ICMP request rules, you minimize these risks while still allowing necessary network communication.

Attack Scenario:

1. Ping Flood Attack:

- An attacker sends a high volume of ICMP Echo Requests (ping requests) to overwhelm the network or a specific target, causing it to become unresponsive. If ICMP traffic is not properly controlled, the target system may be flooded with requests, resulting in degraded performance or complete denial of service.

2. ICMP Redirect Attack:

- An attacker sends ICMP Redirect messages to manipulate the routing of network traffic. This could redirect traffic to an attacker-controlled system, potentially allowing them to intercept or alter communications. If the firewall allows unrestricted ICMP Redirects, attackers can exploit this to compromise network security.

Audit Procedure:

To check and verify the secure configuration of ICMP requests:

1. In the GUI:

- Navigate to `Firewall > Rules`.
- Review the existing firewall rules to identify any that pertain to ICMP requests.
- Check each rule to see the types of ICMP requests allowed (e.g., Echo Request, Echo Reply, Destination Unreachable).
- Ensure that only necessary ICMP types are permitted and that other types are blocked or restricted as appropriate.

Remediation Procedure:

To configure ICMP requests securely:

1. In the GUI:

- Navigate to `Firewall > Rules`.

- Locate and review the rules that allow ICMP traffic.
- Edit each rule to ensure it only allows essential ICMP request types. For example:
 - **Allow ICMP Echo Request:** If you need to permit ping requests for network diagnostics.
 - **Block ICMP Redirect:** To prevent attackers from manipulating routing tables.
- Remove or restrict any rules that allow unnecessary ICMP types.
- Click **Save** to apply changes.

2. Example Configuration:

- Create or modify a firewall rule as follows:
 - **Action:** Pass or Block
 - **Interface:** (Specify the appropriate network interface)
 - **Source:** (Specify the source network or IP address)
 - **Destination:** (Specify the destination network or IP address)
 - **Protocol:** ICMP
 - **ICMP Type:** Select necessary types (e.g., Echo Request, Echo Reply) and block other types (e.g., Redirect).

Default Configuration:

- By default, many firewalls might allow all types of ICMP traffic. Secure configurations typically involve restricting or customizing ICMP request types to balance functionality and security.

If you need to apply ICMP configurations through CLI (assuming you have the capability or are using a system that supports such commands), you might use something similar to the following:

1. Check Current ICMP Rules:

COPY 

```
pfctl -sr | grep icmp
```

Add a Rule to Allow Only Specific ICMP Types:

COPY 

```
# Example for pf (Packet Filter) in FreeBSD-based systems
echo 'pass in proto icmp from any to any icmp-type echoreq' >> /etc/pf.conf
```

Block Unnecessary ICMP Types:

COPY 

```
# Example for pf
echo 'block in proto icmp from any to any icmp-type redirect' >> /etc/pf.conf
```

Reload the Firewall Configuration:

COPY 

```
pfctl -f /etc/pf.conf
pfctl -e
```

SNMP

This control involves ensuring that SNMP (Simple Network Management Protocol) traps are configured and enabled for monitoring network events. SNMP traps are

notifications sent from network devices to a Network Management Station (NMS) to inform administrators of significant events or conditions on the network.

Rationale:

Enabling SNMP traps allows real-time monitoring and alerting of network events, which helps in maintaining the availability and security of network systems. By configuring SNMP traps to be sent to designated receivers, administrators can ensure timely responses to network issues and effectively manage network resources.

Attack Scenario:

1. Lack of Monitoring:

- **Scenario:** If SNMP traps are not enabled or configured, critical network events such as unauthorized access attempts, system failures, or performance issues might go unnoticed. An attacker could exploit this lack of visibility to carry out malicious activities without immediate detection.
- **Impact:** The absence of SNMP traps can lead to delayed responses to security incidents, potentially allowing attackers to cause more damage before any corrective actions are taken.

2. Misconfiguration:

- **Scenario:** Incorrect configuration of SNMP trap settings (e.g., wrong IP address of the NMS or improper SNMP community strings) can prevent traps from being delivered. This misconfiguration can result in critical alerts not reaching the administrators.
- **Impact:** Misconfigured SNMP traps might cause critical events to be missed, leaving the network vulnerable to prolonged exposure and exploitation.

Audit Procedure:

1. Check SNMP Traps Configuration:

- **In the GUI:**

- Navigate to **Services > SNMP**.
- Ensure the **SNMP Traps Enable** checkbox is checked.
- Verify the **SNMP Trap Settings** are correctly configured to point to the appropriate NMS.

2. Verify SNMP Traps Functionality:

- **In the GUI:**
 - Again, navigate to **Services > SNMP**.
 - Confirm that the **SNMP Traps Enable** checkbox is selected.
 - Check that the SNMP Trap settings are correctly set to ensure traps are sent to the NMS.

Remediation Procedure:

1. Enable SNMP Traps and Configure Receivers:

- **In the GUI:**
 - Navigate to **Services > SNMP**.
 - Check the **SNMP Traps Enable** checkbox to activate SNMP traps.
 - Configure the **SNMP Trap Settings**:
 - **Trap Receivers:** Enter the IP address of the Network Management Station (NMS) where traps should be sent.
 - **Community String:** Set the appropriate community string used for SNMP communication.
 - Click **Save** to apply the changes.

2. Example Configuration:

- **Trap Receivers:**
 - **IP Address:** Enter the IP address of your NMS.

- **Port:** Typically, SNMP traps are sent to UDP port 162.
- **Community String:** Ensure it matches what the NMS is configured to expect.
- **Configure SNMP Trap Settings:**
 - Set the trap settings as required for your environment.

If applicable for systems with CLI support for SNMP configuration, use the following commands as an example:

1. Enable SNMP Daemon:

```
# For FreeBSD or similar systems
sysrc snmpd_enable=YES
```

COPY 

2. Edit SNMP Configuration File:

```
# Example for editing the SNMP configuration in /usr/local/etc/snmpd.conf
vi /usr/local/etc/snmpd.conf
# Add or modify the following lines:
trap community public
trap 192.168.1.100
```

COPY 

3. Restart SNMP Service:

```
# Restart SNMP service to apply changes
service snmpd restart
```

COPY 

Default Values:

- **SNMP Traps Enable:** By default, this option may be unchecked, meaning SNMP traps are not active until manually enabled.
- **Trap Receivers:** The default configuration typically does not have any receivers set up until specified by the user.

Resources

- CIS v1.1.0

Subscribe to our newsletter

Read articles from **DevSecOpsGuides** directly inside your inbox. Subscribe to the newsletter, and don't miss out.

reza.rashidi.business@gmail.com

SUBSCRIBE

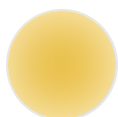
Devops

DevSecOps


pfsense

firewall

Written by



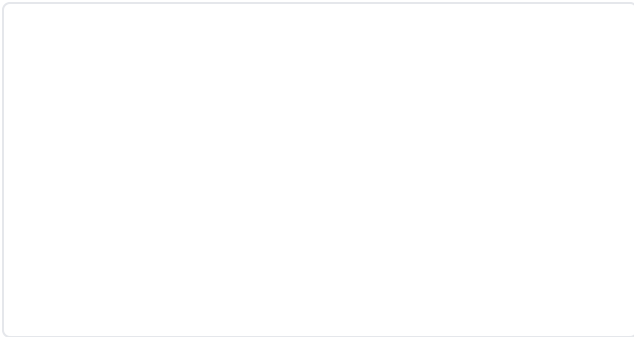
Reza Rashidi

 Add your bio



MORE ARTICLES

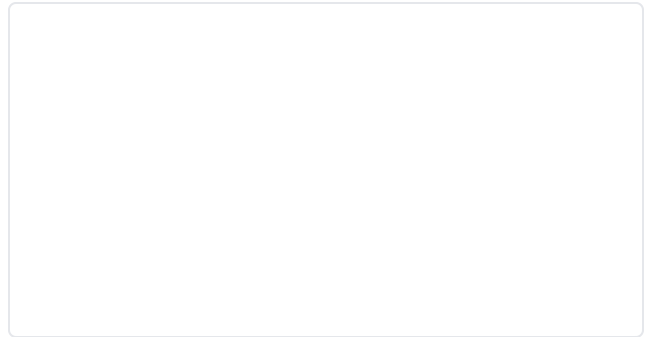
Reza Rashidi



Attacking Nginx

Nginx, a popular web server and reverse proxy, is a critical component in many web infrastructures, ...

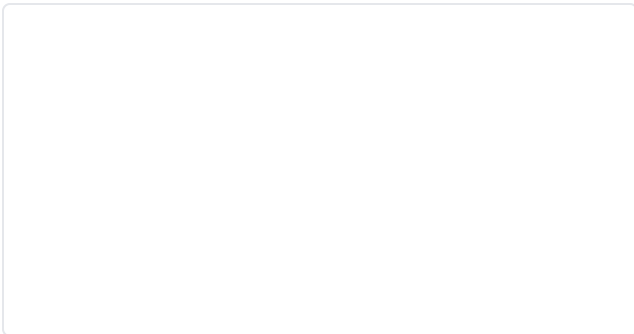
Reza Rashidi



Attacking OpenStack

Attacking OpenStack, an open-source cloud computing platform, involves exploiting vulnerabilities in...

Reza Rashidi



Attacking CI/CD

In CI/CD (Continuous Integration/Continuous Deployment) environments, several methods and attacks ca...

©2024 DevSecOpsGuides

[Archive](#) · [Privacy_policy](#) · [Terms](#)



Write on Hashnode

Powered by [Hashnode](#) - Home for tech writers and readers