2023

Vulnerability Report

WWW.HAVMOR.COM

TEAM 1.2

SIVARAGHAVI U R (21BRS1412)

SARATH RAJAN SENTHILKUMAR (21BCE5965)

Note: Changed the main website form wikipedia.org to havmor.com

VULNERABILITY NAME: SSH Weak Key Exchange Algorithms Enabled

CWE: CWE-326

OWASP Category: A02:2021-Cryptographic Failures.

Description: The vulnerability "SSH Weak Key Exchange Algorithms Enabled" signifies a security concern where the Secure Shell protocol is configured to allow the use of outdated and potentially exploitable cryptographic algorithms during key exchange. This weakness could expose systems to the risk of unauthorized access or data compromise.

Business Impact: The business impact of "SSH Weak Key Exchange Algorithms Enabled" can be severe, leading to unauthorized access, data breaches, and potential compromise of sensitive information. Such vulnerabilities could result in financial losses, damage to reputation, and legal consequences.

Vulnerability Path: https://www.havmor.com/

Vulnerability Parameter: https://www.havmor.com/

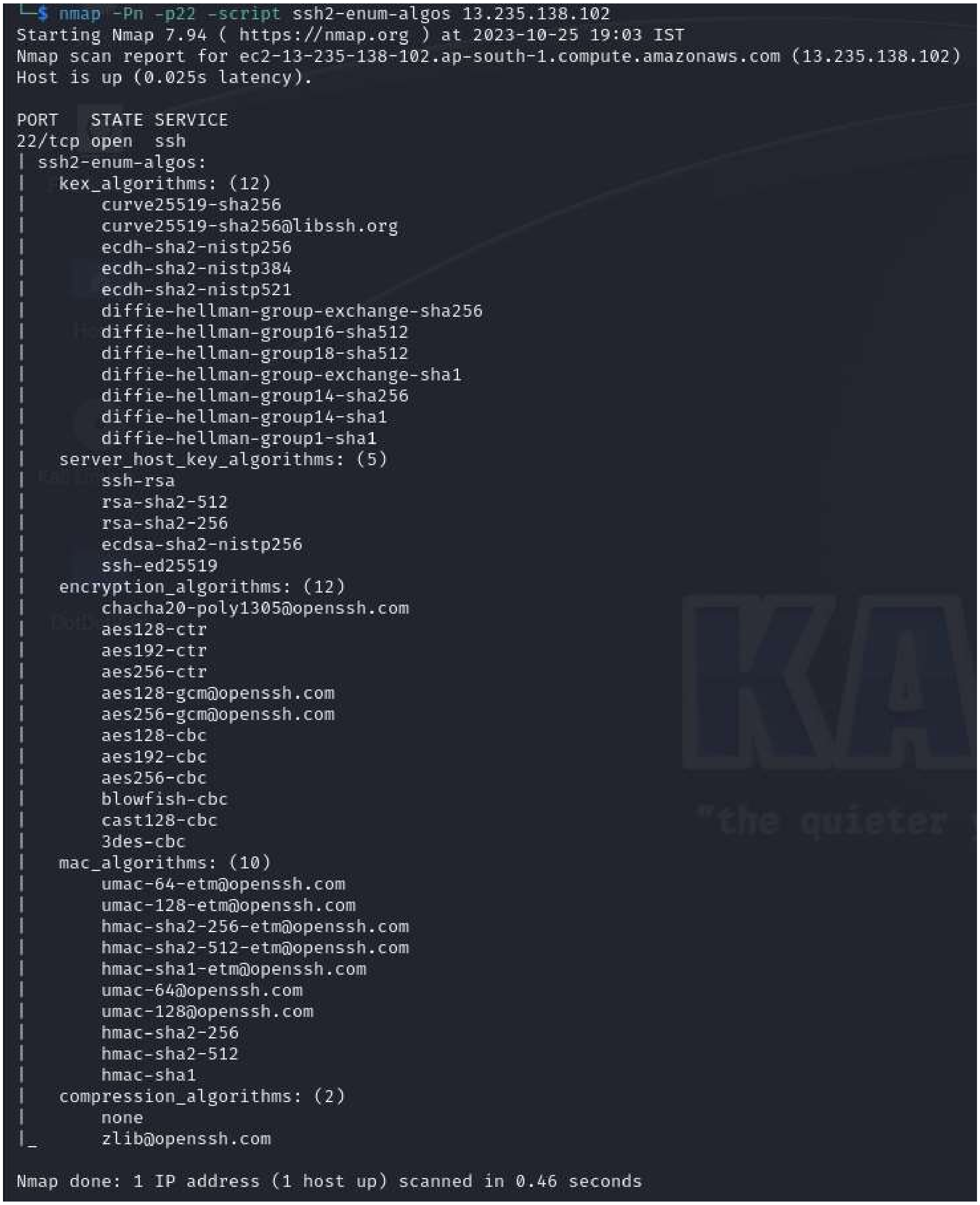
Steps to Reproduce:

Step 1: Open a Kali Linux Terminal.

Step 2: Run the following command:

```
nmap -Pn -p22 -script ssh2-enum-algos 13.235.138.102
```

Note: 13.235.138.102 is the IPV4 Address for www.havmor.com.



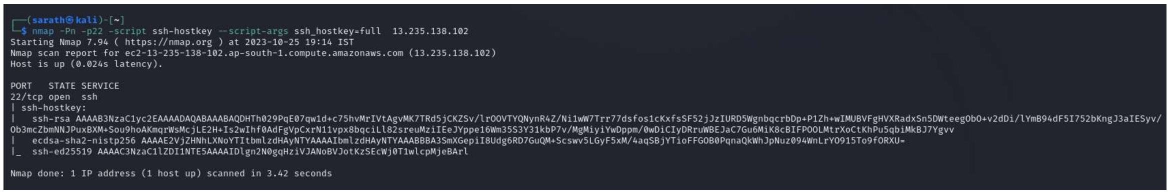
```
└─$ nmap -Pn -p22 -script ssh2-enum-algos 13.235.138.102
Starting Nmap 7.94 ( https://nmap.org ) at 2023-10-25 19:03 IST
Nmap scan report for ec2-13-235-138-102.ap-south-1.compute.amazonaws.com (13.235.138.102)
Host is up (0.025s latency).

PORT   STATE SERVICE
22/tcp open  ssh
| ssh2-enum-algos:
|   kex_algorithms: (12)
|       curve25519-sha256
|       curve25519-sha256@libssh.org
|       ecdh-sha2-nistp256
|       ecdh-sha2-nistp384
|       ecdh-sha2-nistp521
|       diffie-hellman-group-exchange-sha256
|       diffie-hellman-group16-sha512
|       diffie-hellman-group18-sha512
|       diffie-hellman-group-exchange-sha1
|       diffie-hellman-group14-sha256
|       diffie-hellman-group14-sha1
|       diffie-hellman-group1-sha1
|   server_host_key_algorithms: (5)
|       ssh-rsa
|       rsa-sha2-512
|       rsa-sha2-256
|       ecdsa-sha2-nistp256
|       ssh-ed25519
|   encryption_algorithms: (12)
|       chacha20-poly1305@openssh.com
|       aes128-ctr
|       aes192-ctr
|       aes256-ctr
|       aes128-gcm@openssh.com
|       aes256-gcm@openssh.com
|       aes128-cbc
|       aes192-cbc
|       aes256-cbc
|       blowfish-cbc
|       cast128-cbc
|       3des-cbc
|   mac_algorithms: (10)
|       umac-64-etm@openssh.com
|       umac-128-etm@openssh.com
|       hmac-sha2-256-etm@openssh.com
|       hmac-sha2-512-etm@openssh.com
|       hmac-sha1-etm@openssh.com
|       umac-64@openssh.com
|       umac-128@openssh.com
|       hmac-sha2-256
|       hmac-sha2-512
|       hmac-sha1
|   compression_algorithms: (2)
|       none
|_      zlib@openssh.com

Nmap done: 1 IP address (1 host up) scanned in 0.46 seconds
```

To find the weak algorithms, run the following command:

nmap -Pn -p22 -script ssh-hostkey --script-args ssh_hostkey=full 13.235.138.102



```
┌──(sarath㉿kali)-[~]
└─$ nmap -Pn -p22 -script ssh-hostkey --script-args ssh_hostkey=full  13.235.138.102
Starting Nmap 7.94 ( https://nmap.org ) at 2023-10-25 19:14 IST
Nmap scan report for ec2-13-235-138-102.ap-south-1.compute.amazonaws.com (13.235.138.102)
Host is up (0.024s latency).

PORT   STATE SERVICE
22/tcp open  ssh
| ssh-hostkey:
|   ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQDHTh029PqE07qw1d+c75hvMrIVtAgvMK7TRd5jCKZSv/lrOOVTYQNynR4Z/Ni1wW7Trr77dsfos1cKxfsSF52jJzIURD5WgnbqcrbDp+P1Zh+wIMUBVFgHVXRadxSn5DWteegObO+v2dDi/lYmB94dF5I752bKngJ3aIESyv/
Ob3mcZbmNNJPuxBXM+Sou9hoAKmqrWsMcjLE2H+Is2wIhf0AdFgVpCxrN11vpx8bqciLl82sreuMziIEeJYppe16Wm35S3Y31kbP7v/MgMiyiYwDppm/0wDiCIyDRruWBEJaC7Gu6MiK8cBIFPOOLMtrXoCtKhPu5qbiMkBJ7Ygvv
|   ecdsa-sha2-nistp256 AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAAIbmlzdHAyNTYAAABBBA3SmXGepiI8Udg6RD7GuQM+Scswv5LGyF5xM/4aqSBjYTioFFGOB0PqnaQkWhJpNuz094WnLrYO915To9fORXU=
|_  ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIDlgn2N0gqHziVJANoBVJotKzSEcWj0T1wlcpMjeBArl

Nmap done: 1 IP address (1 host up) scanned in 3.42 seconds
```

Recommendations:

- Disable Weak Algorithms.
- Update Configurations.
- Establish Monitoring and Auditing

VULNERABILITY NAME: SSH Server CBC Mode Ciphers Enabled

CWE: CWE-327

OWASP Category: A02:2021-Cryptographic Failures.

Description: The vulnerability "SSH Server CBC Mode Ciphers Enabled" indicates that the Cipher Block Chaining (CBC) mode ciphers are active in the configuration of the Secure Shell (SSH) server. CBC is a block cipher mode widely used in cryptographic protocols, but it is known to have certain vulnerabilities, particularly related to information leakage and padding oracle attacks. When CBC mode ciphers are enabled in the SSH server, it poses a potential risk of exploitation by attackers.

Business Impact: Enabling "SSH Server CBC Mode Ciphers" poses a significant business impact by exposing organizations to data confidentiality risks and potential unauthorized access. This vulnerability may lead to regulatory compliance concerns, attracting legal consequences and fines for failing to meet industry standards. Additionally, the exploitation of insecure SSH configurations could damage the organization's reputation and result in operational disruptions, emphasizing the critical need for prompt mitigation measures.

Vulnerability Path: https://www.havmor.com/

Vulnerability Parameter: https://www.havmor.com/

Steps to Reproduce:

Step 1: Open a Kali Linux Terminal.

Step 2: Run the following command.

```
nmap --script ssl-enum-ciphers -p 443,465,993,995 13.235.138.102
```

Note: 13.235.138.102 is the IPV4 Address for www.havmor.com.

Recommendations:

- Disable CBC Mode Ciphers.
- Update SSH Configuration.
- Conduct Security Audits.

VULNERABILITY NAME: HSTS Missing from HTTPS Server

CWE: CWE-523

OWASP Category: A05:2021-Security Misconfiguration

Description: The absence of HTTP Strict Transport Security (HSTS) from an HTTPS server represents a security vulnerability where the server fails to enforce a policy instructing web browsers to communicate exclusively over secure HTTPS connections. HSTS is a critical security mechanism designed to mitigate the risk of man-in-the-middle attacks and enhance the overall security posture of web applications.

Business Impact: The business impact of HSTS missing from an HTTPS server includes heightened security risks, increasing the vulnerability to man-in-the-middle attacks and potential unauthorized access to sensitive data. This absence may lead to data breaches, erode user trust, and result in regulatory non-compliance, impacting the organization's reputation and exposing it to legal consequences. Without HSTS, there is a risk of session hijacking, potentially causing financial losses and putting the business at a competitive disadvantage in a security-conscious market.

Vulnerability Path: https://www.havmor.com/

Vulnerability Parameter: https://www.havmor.com/

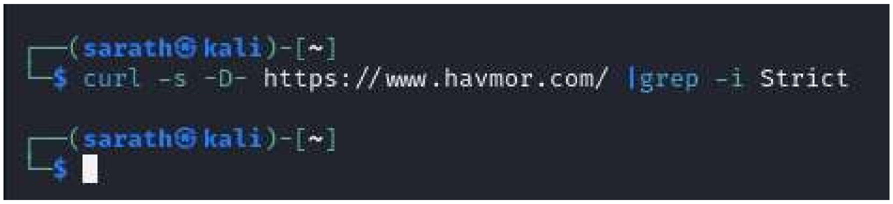
Steps to Reproduce:

Step 1: Open a Kali Linux Terminal.

Step 2: Run the following command.

```
curl -s -D- https://www.havmor.com/ |grep -i Strict
```

Note: If HSTS is present a message would follow following the execution of the command. If HSTS is not present no message would follow following the execution of the command.



```
┌──(sarath㉿kali)-[~]
└─$ curl -s -D- https://www.havmor.com/ |grep -i Strict

┌──(sarath㉿kali)-[~]
└─$ 
```

Recommendations:

- Implement HSTS Header.
- Configure "includeSubDomains" if Applicable.
- Consider "preload" and Submit to HSTS Preload Lists.

VULNERABILITY NAME: DNS Server Spoofed Request Amplification DDoS

CWE: CWE-406

OWASP Category: A04:2021-Insecure Design.

Description: The "DNS Server Spoofed Request Amplification DDoS" vulnerability allows attackers to exploit weaknesses in DNS servers by sending malicious requests with spoofed source IPs. By leveraging the DNS protocol's amplification characteristics, the attackers generate large responses to small requests, overw3helming the target with a flood of traffic and causing a Distributed Denial of Service (DDoS) scenario.

Business Impact: A "DNS Server Spoofed Request Amplification DDoS" attack can inflict severe business impact, causing service disruptions, financial losses, and reputational damage. Downtime and unavailability may lead to dissatisfied customers, legal consequences, and increased operational costs for mitigating and recovering from the attack. Businesses may also face a competitive disadvantage as resilient competitors gain market share during disruptive periods, emphasizing the critical need for robust DDoS mitigation strategies.

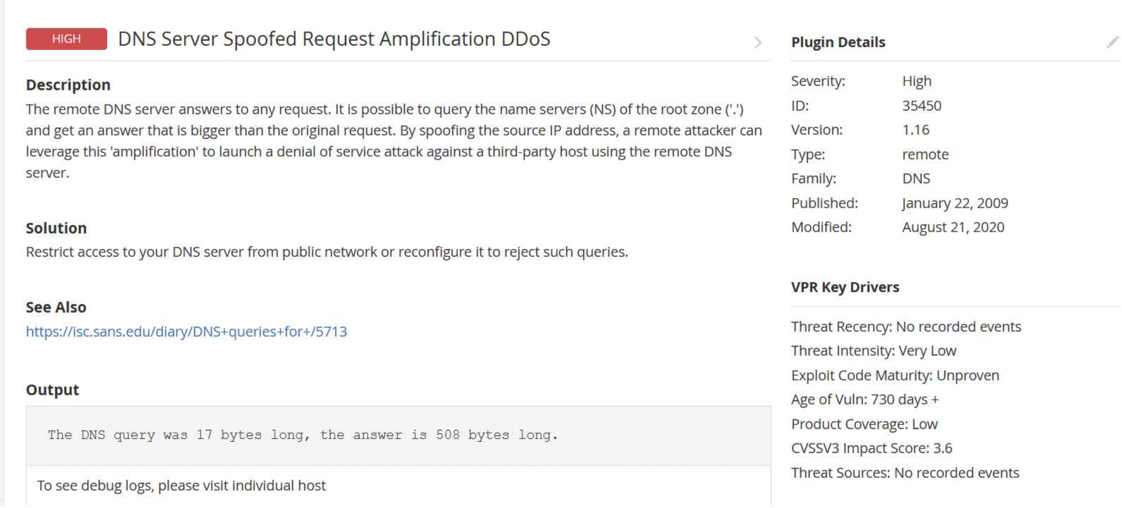
Vulnerability Path: https://www.havmor.com/

Vulnerability Parameter: https://www.havmor.com/

Steps to Reproduce:

Step 1: Run a Basic Network Scan for the IP Address of www.havmor.com.

Step 2: Check the vulnerability page for the Scan.



HIGH DNS Server Spoofed Request Amplification DDoS

Description

The remote DNS server answers to any request. It is possible to query the name servers (NS) of the root zone ('.') and get an answer that is bigger than the original request. By spoofing the source IP address, a remote attacker can leverage this 'amplification' to launch a denial of service attack against a third-party host using the remote DNS server.

Solution

Restrict access to your DNS server from public network or reconfigure it to reject such queries.

See Also

https://isc.sans.edu/diary/DNS+queries+for+/5713

Output

```
The DNS query was 17 bytes long, the answer is 508 bytes long.
```

To see debug logs, please visit individual host

Plugin Details

Severity:	High
ID:	35450
Version:	1.16
Type:	remote
Family:	DNS
Published:	January 22, 2009
Modified:	August 21, 2020

VPR Key Drivers

Threat Recency: No recorded events
Threat Intensity: Very Low
Exploit Code Maturity: Unproven
Age of Vuln: 730 days +
Product Coverage: Low
CVSSV3 Impact Score: 3.6
Threat Sources: No recorded events

Recommendations:

- Implement Rate Limiting.
- Update DNS Software.
- Enable Response Rate Limiting (RRL).

VULNERABILITY NAME: DNS Server Recursive Query Cache Poisoning Weakness

CWE: CWE-682

OWASP Category: A05:2021-Security Misconfiguration

Description: DNS Server Recursive Query Cache Poisoning is a security weakness where an attacker exploits vulnerabilities in a DNS (Domain Name System) server's handling of recursive queries. In this scenario, an adversary can manipulate the responses to these queries, injecting fraudulent DNS records into the server's cache.

Business Impact: DNS Server Recursive Query Cache Poisoning can disrupt services, compromise sensitive data, and erode customer trust by redirecting legitimate traffic to malicious destinations. The financial consequences include operational costs for remediation, potential legal repercussions, and a competitive disadvantage in the market. This vulnerability necessitates urgent mitigation to prevent business impacts such as downtime, data breaches, and damage to reputation and customer relationships.

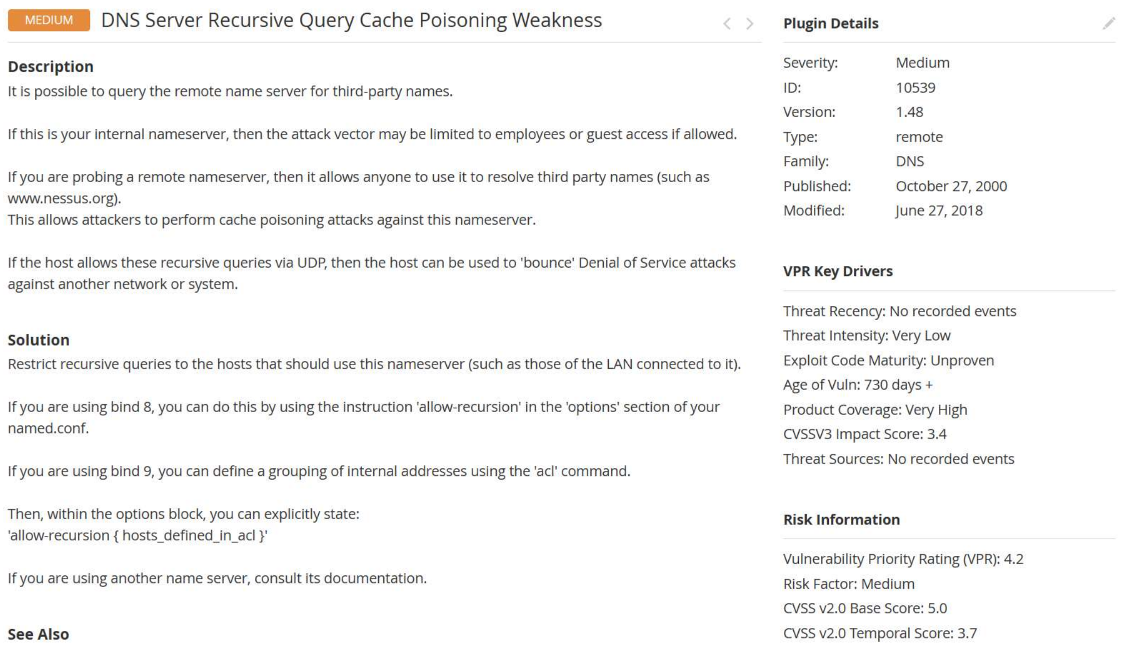
Vulnerability Path: https://www.havmor.com/

Vulnerability Parameter: https://www.havmor.com/

Steps to Reproduce:

Step 1: Run a Basic Network Scan for the IP Address of www.havmor.com.

Step 2: Check the vulnerability page for the Scan.



MEDIUM DNS Server Recursive Query Cache Poisoning Weakness

Description

It is possible to query the remote name server for third-party names.

If this is your internal nameserver, then the attack vector may be limited to employees or guest access if allowed.

If you are probing a remote nameserver, then it allows anyone to use it to resolve third party names (such as www.nessus.org).
This allows attackers to perform cache poisoning attacks against this nameserver.

If the host allows these recursive queries via UDP, then the host can be used to 'bounce' Denial of Service attacks against another network or system.

Solution

Restrict recursive queries to the hosts that should use this nameserver (such as those of the LAN connected to it).

If you are using bind 8, you can do this by using the instruction 'allow-recursion' in the 'options' section of your named.conf.

If you are using bind 9, you can define a grouping of internal addresses using the 'acl' command.

Then, within the options block, you can explicitly state:
'allow-recursion { hosts_defined_in_acl }'

If you are using another name server, consult its documentation.

See Also

Plugin Details

Severity:	Medium
ID:	10539
Version:	1.48
Type:	remote
Family:	DNS
Published:	October 27, 2000
Modified:	June 27, 2018

VPR Key Drivers

Threat Recency: No recorded events
Threat Intensity: Very Low
Exploit Code Maturity: Unproven
Age of Vuln: 730 days +
Product Coverage: Very High
CVSSV3 Impact Score: 3.4
Threat Sources: No recorded events

Risk Information

Vulnerability Priority Rating (VPR): 4.2
Risk Factor: Medium
CVSS v2.0 Base Score: 5.0
CVSS v2.0 Temporal Score: 3.7

Recommendations:

- Implement DNS Response Validation.
- Update and Patch DNS Software.
- Use DNS Security Extension (DNSSEC).