MGS 650 Information Assurance
Lab 4: Vulnerability Scanning and Management
Submitted by: Akhilesh Anand Undralla
11/26/2021

Summary

This vulnerability scan by OpenVAS started at Wed Nov 24 07:17:39 2021 UTC and ended at Wed Nov 24 07:40:36 2021 UTC reports the results of an immediate security scan IP 192.168.252.3, 192.168.252.61, 192.168.252.115. The scan provides the results found for each host. Then, for each host, the report describes every issue found. Refer Figure [1.0].

The vulnerabilities/lack of controls found from three machines are as follows:

- Vendor security updates are not trusted.
- Overrides are on.
- When a result has an override, this report uses the threat of the override.
- Information on overrides is included in the report.

Vulnerabilities related to MySQL / MariaDB, SSH / OpenSSH, TCH Timestamps were found on these machines. Refer Figure [1.4]. The solutions to above vulnerabilities include mitigation, vendor fixes, workaround to restrict access and upgrades. While 192.168.252.115 machine has no reported vulnerabilities, 192.168.252.3 & 192.168.252.61 reporoted 4 results out of 59 results where 41 results are filtered out based on severity/risk factor.

MySQL / MariaDB Vulnerabilities:

The host IP 192.168.252.61 has vulnerabilities related to MySQL / MariaDB in which MariaDB version on the remote host has reached the end of life (Installed version: 5.5.56, EOL version: 5.5) and should not be used anymore. The MariaDB version installed is not receiving any security updates from the vendor. This results in unfixed patches that might be leveraged by an attacker to compromise the security of this host. Refer Figure [1.5]. The solution is to update the MariaDB version on the remote host to a still supported version. This vulnerability is considered as high severity with Common Vulnerability Scoring System (CVSS) of 10.0

Another vulnerability related to MySQL / MariaDB is weak credentials to login into the remote MySQL as root. Here, it is possible to login as root with an empty password. This must be mitigated by changing the password immediately. This vulnerability is considered as high severity with Common Vulnerability Scoring System (CVSS) of 9.0

A vulnerability related to MariaDB is the host 192.168.252.61 running on MariaDB prone to an access bypass vulnerability that has a medium severity with CVSS of 6.5. Any user with SQL access to the server could possibly use this vulnerability to perform database modification on certain cluster nodes without having privileges to perform such changes. The solution is to update the MariaDB version from 5.5.56 (current) to 10.1.30, 10.2.10 or later. Quality of Detection (QoD) for vulnerability is determined as 95%, where QoD is a value between 0-100 % that describes the reliability of an executed vulnerability detection/product detection.

SSH Vulnerabilities:

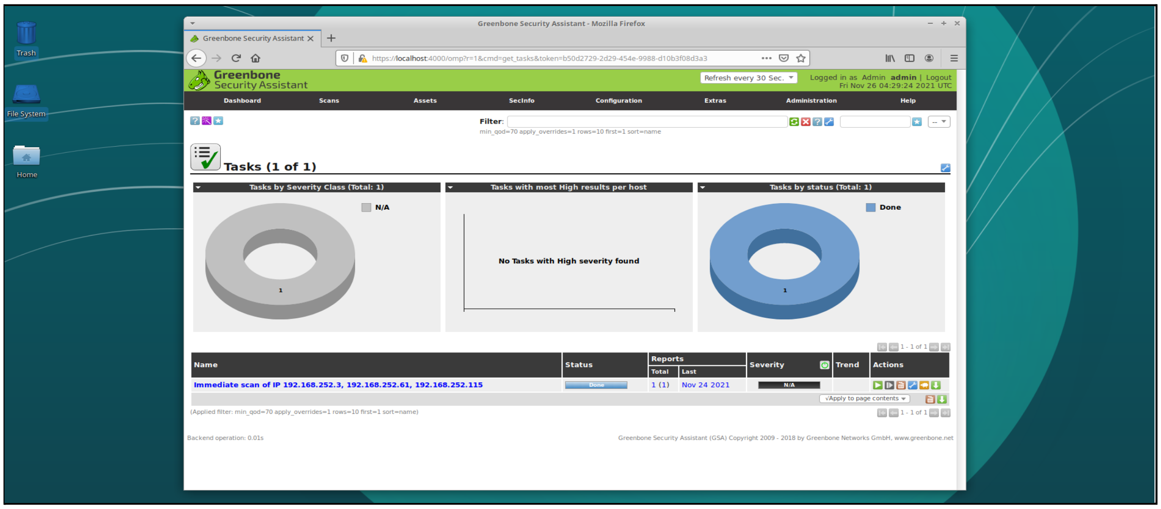
The host IP 192.168.252.3 has vulnerabilities related to SSH where remote SSH server configured to allow weak encryption algorithms including both client-to-server and server-to-client weak encryption algorithms. The algorithms used are `arcfour` cipher 128-bits and a `none` algorithm which specifies that no encryption is to be done which is not recommended since it provides no confidentiality protection. Another vulnerability exists in SSH messages that employ Cipher block chaining (CBC) mode of attacks that may allow an attacker to recover plaintext from a block of ciphertext. The solution is to mitigate by disabling the weak encryption algorithms on the remote ssh service that supports Arcfour, none or CBC ciphers.. This vulnerability has medium severity with CVSS of 4.3 and QoD as 95%. Refer Figure [1.6].

Few other medium (CVSS: 5.0) to High (CVSS: 8.5) severity ranging OpenSSH vulnerabilities are user enumeration vulnerability, security bypass vulnerability, client information leaks, impersonation attacks, denial of service and xauth command injection. User enumeration vulnerability is where remote attackers test whether a certain user exists or not (username enumeration) on a target OpenSSH server, harvest valid user accounts and perform brute-force attacks. Secure Bypass vulnerability allows local users to bypass certain security restrictions and perform unauthorized actions. Denial of service attacks (out-of-bounds read and application crash). This flaw exists due to an error in 'ssh_packet_read_poll2' function with in 'packet.c' script. Xauth command injection may lead to forced-command, gains limited* read/write arbitrary files, information leakage and /bin/false bypass. This is done by injecting xauth commands by an authenticated user by sending an x11 channel request that includes a newline character in the x11 cookie and the newline acts as a command separator to the xauth binary. This attack requires the server to have 'X11Forwarding yes' enabled. The solution is to mitigate this vector by updating OpenSSH-6.6.1 to 7.2 or later as well as disabling X11 forwarding.

TCP Timestamp Vulnerabilities:

The host IP 192.168.252.3 and 192.168.252.61 has vulnerabilities related to TCP timestamps feature where the uptime of the remote host that implements TCP timestamps can be computed by retrieving timestamp packets. The solution to this vulnerability is to mitigate this by disabling TCP timestamps on linux by adding the line 'net.ipv4.tcp_timestamps = 0' to /etc/sysctl.conf. After that, execute 'sysctl -p' to apply the configuration settings at runtime. On Windows, to disable TCP timestamps, execute 'netsh int tcp set global timestamps=disabled'. The default behavior of the TCP/IP stack on few legacy systems is to not use the timestamp options when initiating TCP connections, but use them if the TCP peer that is initiating communication includes them in their synchronize (SYN) segment. This vulnerability is considered as Low severity with Common Vulnerability Scoring System (CVSS) of 2.6 and Quality of Detection (QoD) as 80%. Refer Figure [1.7 & 1.8]

Appendix:


Trash
File System
Home
Greenbone Security Assistant - Mozilla Firefox
Greenbone Security Assistant
Greenbone
Security Assistant
Refresh every 30 Sec.
Dashboard
Scans
Assets
SecInfo
Configuration
Extras
Administration
Help
Tasks by Severity Class (Total: 1)
N/A
1
Tasks with most High results per host
No Tasks with High severity found
Tasks by status (Total: 1)
Done
1
1 - 1 of 1
√Apply to page contents
1 - 1 of 1

https://localhost:4000/omp?r=1&cmd=get_tasks&token=b50d2729-2d29-454e-9988-d10b3f08d3a3

Logged in as Admin **admin** | Logout
Fri Nov 26 04:29:24 2021 UTC

Filter:
min_qod=70 apply_overrides=1 rows=10 first=1 sort=name

Tasks (1 of 1)

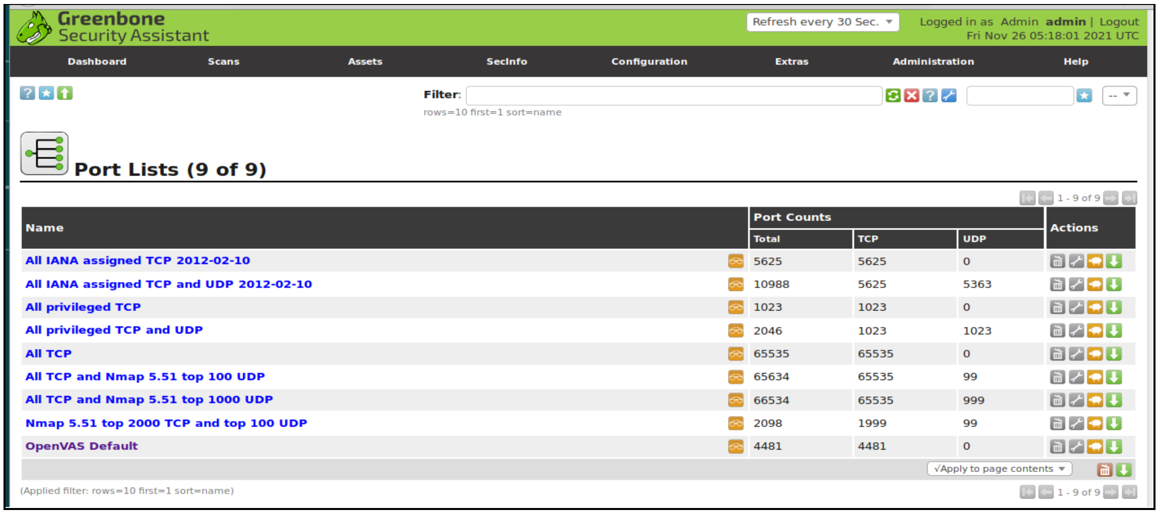
Name	Status	Reports		Severity	Trend	Actions
		Total	Last			
Immediate scan of IP 192.168.252.3, 192.168.252.61, 192.168.252.115	Done	1 (1)	Nov 24 2021	N/A		

(Applied filter: min_qod=70 apply_overrides=1 rows=10 first=1 sort=name)

Backend operation: 0.01s

Greenbone Security Assistant (GSA) Copyright 2009 - 2018 by Greenbone Networks GmbH, www.greenbone.net

Figure 1.0: Tasks with “Immediate scan of IP 192.168.252.3, 192.168.252.61, 192.168.252,115”



Greenbone Security Assistant

Refresh every 30 Sec. ▾ Logged in as Admin **admin** | Logout
Fri Nov 26 05:18:01 2021 UTC

Filter:
rows=10 first=1 sort=name

Port Lists (9 of 9)

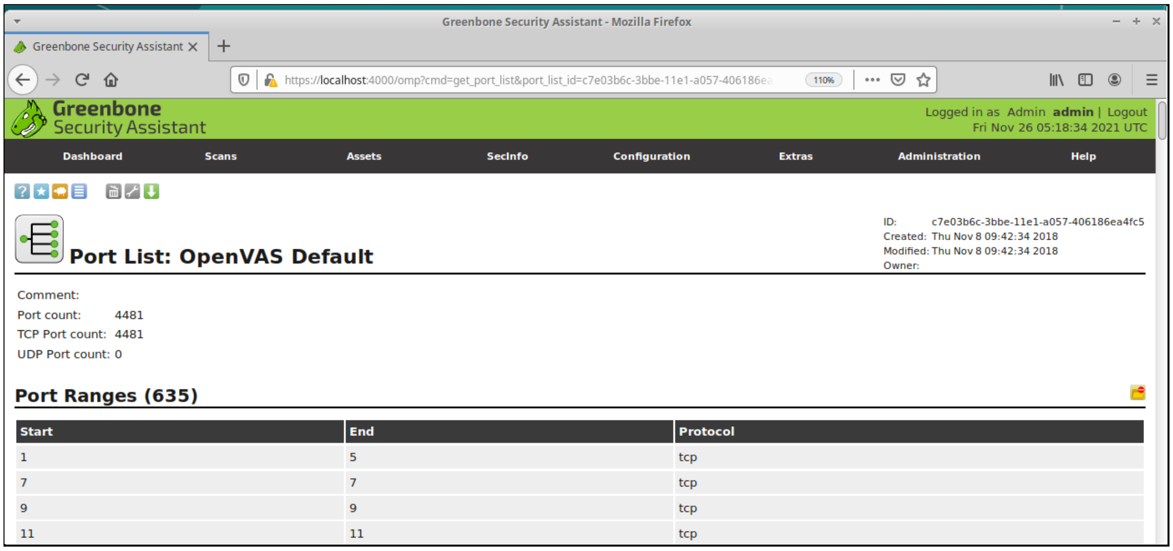
1 - 9 of 9

Name	Port Counts			Actions
	Total	TCP	UDP	
All IANA assigned TCP 2012-02-10	5625	5625	0	
All IANA assigned TCP and UDP 2012-02-10	10988	5625	5363	
All privileged TCP	1023	1023	0	
All privileged TCP and UDP	2046	1023	1023	
All TCP	65535	65535	0	
All TCP and Nmap 5.51 top 100 UDP	65634	65535	99	
All TCP and Nmap 5.51 top 1000 UDP	66534	65535	999	
Nmap 5.51 top 2000 TCP and top 100 UDP	2098	1999	99	
OpenVAS Default	4481	4481	0	

√Apply to page contents ▾

(Applied filter: rows=10 first=1 sort=name) 1 - 9 of 9

Figure 1.1: Port lists that provide configurations listing ports within scope of scan



Greenbone Security Assistant - Mozilla Firefox

Greenbone Security Assistant

https://localhost:4000/omp?cmd=get_port_list&port_list_id=c7e03b6c-3bbe-11e1-a057-406186ea

Greenbone Security Assistant

Logged in as Admin **admin** | Logout
Fri Nov 26 05:18:34 2021 UTC

Port List: OpenVAS Default

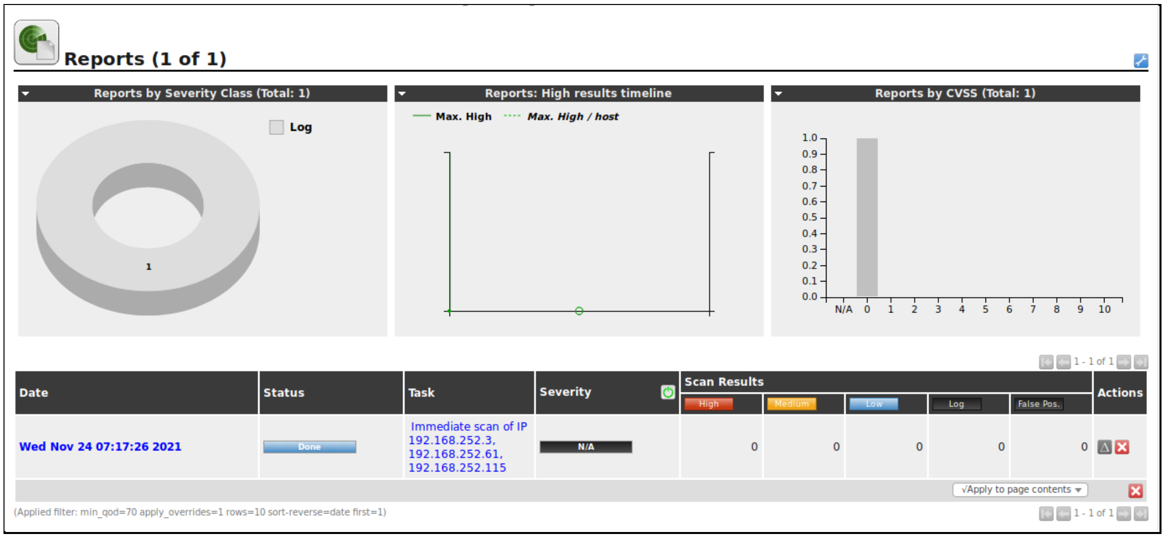
ID: c7e03b6c-3bbe-11e1-a057-406186ea4fc5
Created: Thu Nov 8 09:42:34 2018
Modified: Thu Nov 8 09:42:34 2018
Owner:

Comment:
Port count: 4481
TCP Port count: 4481
UDP Port count: 0

Port Ranges (635)

Start	End	Protocol
1	5	tcp
7	7	tcp
9	9	tcp
11	11	tcp

Figure 1.2: Default port list with OpenVAS Default

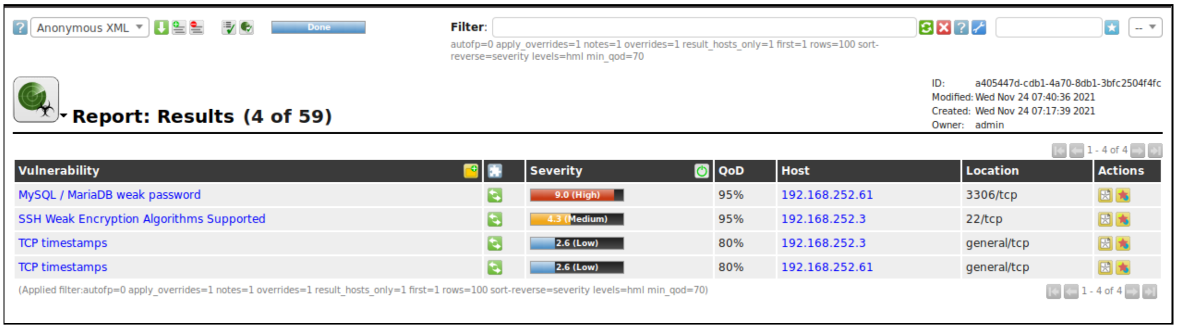


Date	Status	Task	Severity	Scan Results					Actions
				High	Medium	Low	Log	False Pos.	
Wed Nov 24 07:17:26 2021	Done	Immediate scan of IP 192.168.252.3, 192.168.252.61, 192.168.252.115	N/A	0	0	0	0	0	

√Apply to page contents

(Applied filter: min_qod=70 apply_overrides=1 rows=10 sort-reverse=date first=1)

Figure 1.3: Overview of ‘Reports’ page (Scans > Reports)



Filter:

autofp=0 apply_overrides=1 notes=1 overrides=1 result_hosts_only=1 first=1 rows=100 sort-reverse=severity levels=hml min_qod=70

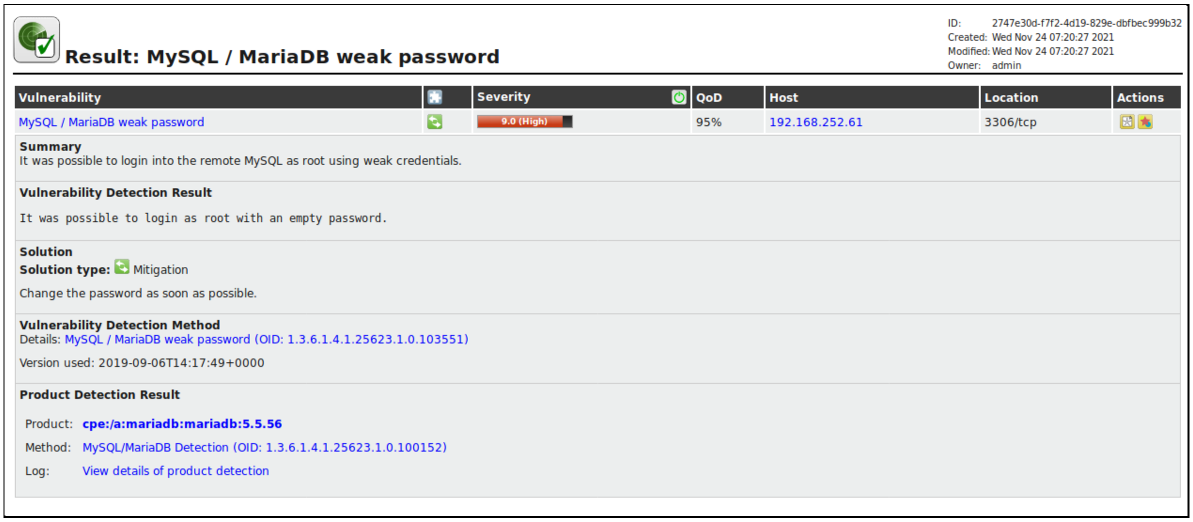
Report: Results (4 of 59)

ID: a405447d-cdb1-4a70-8db1-3bfc2504f4fc
Modified: Wed Nov 24 07:40:36 2021
Created: Wed Nov 24 07:17:39 2021
Owner: admin

Vulnerability		Severity	QoD	Host	Location	Actions
MySQL / MariaDB weak password		9.0 (High)	95%	192.168.252.61	3306/tcp	
SSH Weak Encryption Algorithms Supported		4.3 (Medium)	95%	192.168.252.3	22/tcp	
TCP timestamps		2.6 (Low)	80%	192.168.252.3	general/tcp	
TCP timestamps		2.6 (Low)	80%	192.168.252.61	general/tcp	

(Applied filter:autofp=0 apply_overrides=1 notes=1 overrides=1 result_hosts_only=1 first=1 rows=100 sort-reverse=severity levels=hml min_qod=70)

Figure 1.4: Overview of ‘Results’ of all the vulnerabilities



Result: MySQL / MariaDB weak password

ID: 2747e30d-f7f2-4d19-829e-dbfbec999b32
Created: Wed Nov 24 07:20:27 2021
Modified: Wed Nov 24 07:20:27 2021
Owner: admin

Vulnerability		Severity	QoD	Host	Location	Actions
MySQL / MariaDB weak password		9.0 (High)	95%	192.168.252.61	3306/tcp	

Summary
It was possible to login into the remote MySQL as root using weak credentials.

Vulnerability Detection Result

```
It was possible to login as root with an empty password.
```

Solution
Solution type: Mitigation

Change the password as soon as possible.

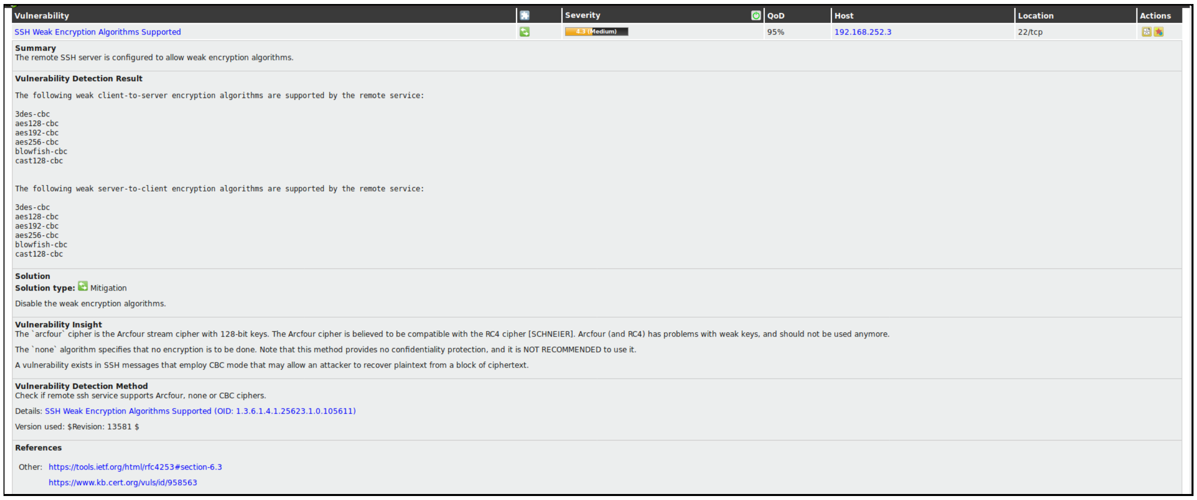
Vulnerability Detection Method
Details: MySQL / MariaDB weak password (OID: 1.3.6.1.4.1.25623.1.0.103551)

Version used: 2019-09-06T14:17:49+0000

Product Detection Result

Product: **cpe:/a:mariadb:mariadb:5.5.56**
Method: MySQL/MariaDB Detection (OID: 1.3.6.1.4.1.25623.1.0.100152)
Log: View details of product detection

Figure 1.5: Overview of ‘MySQL / MariaDB Vulnerability’



Vulnerability		Severity	QoD	Host	Location	Actions
SSH Weak Encryption Algorithms Supported		4.3 (Medium)	95%	192.168.252.3	22/tcp	

Summary
The remote SSH server is configured to allow weak encryption algorithms.

Vulnerability Detection Result

```
The following weak client-to-server encryption algorithms are supported by the remote service:

3des-cbc
aes128-cbc
aes192-cbc
aes256-cbc
blowfish-cbc
cast128-cbc


The following weak server-to-client encryption algorithms are supported by the remote service:

3des-cbc
aes128-cbc
aes192-cbc
aes256-cbc
blowfish-cbc
cast128-cbc
```

Solution
Solution type: Mitigation

Disable the weak encryption algorithms.

Vulnerability Insight
The `arcfour` cipher is the Arcfour stream cipher with 128-bit keys. The Arcfour cipher is believed to be compatible with the RC4 cipher [SCHNEIER]. Arcfour (and RC4) has problems with weak keys, and should not be used anymore.

The `none` algorithm specifies that no encryption is to be done. Note that this method provides no confidentiality protection, and it is NOT RECOMMENDED to use it.

A vulnerability exists in SSH messages that employ CBC mode that may allow an attacker to recover plaintext from a block of ciphertext.

Vulnerability Detection Method
Check if remote ssh service supports Arcfour, none or CBC ciphers.

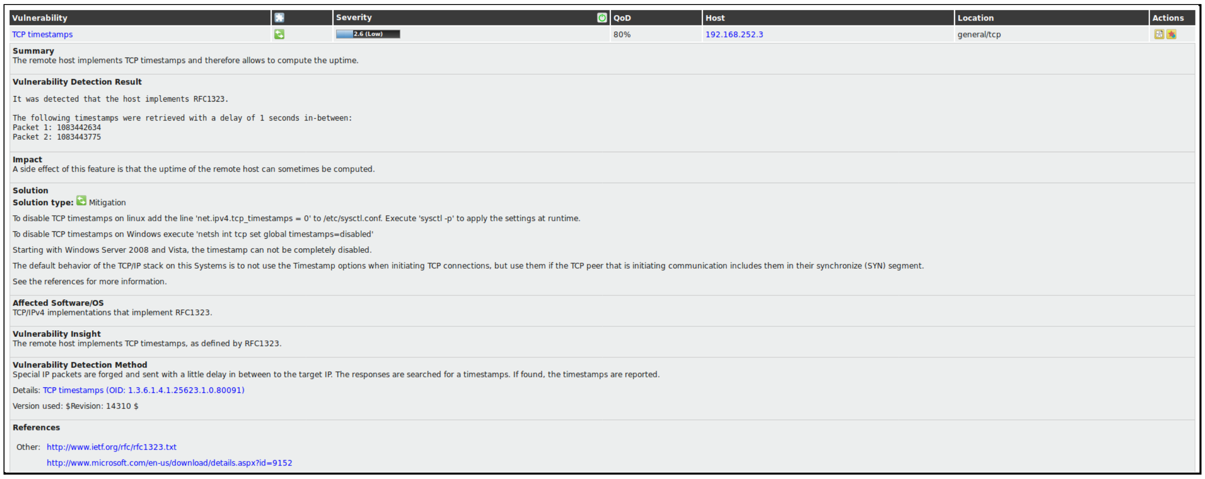
Details: SSH Weak Encryption Algorithms Supported (OID: 1.3.6.1.4.1.25623.1.0.105611)

Version used: $Revision: 13581 $

References

Other: https://tools.ietf.org/html/rfc4253#section-6.3
https://www.kb.cert.org/vuls/id/958563

Figure 1.6: Overview of ‘SSH Vulnerability’



Vulnerability		Severity	QoD	Host	Location	Actions
TCP timestamps		2.6 (Low)	80%	192.168.252.3	general/tcp	

Summary
The remote host implements TCP timestamps and therefore allows to compute the uptime.

Vulnerability Detection Result

```
It was detected that the host implements RFC1323.

The following timestamps were retrieved with a delay of 1 seconds in-between:
Packet 1: 1083442634
Packet 2: 1083443775
```

Impact
A side effect of this feature is that the uptime of the remote host can sometimes be computed.

Solution
Solution type: Mitigation

To disable TCP timestamps on linux add the line 'net.ipv4.tcp_timestamps = 0' to /etc/sysctl.conf. Execute 'sysctl -p' to apply the settings at runtime.

To disable TCP timestamps on Windows execute 'netsh int tcp set global timestamps=disabled'

Starting with Windows Server 2008 and Vista, the timestamp can not be completely disabled.

The default behavior of the TCP/IP stack on this Systems is to not use the Timestamp options when initiating TCP connections, but use them if the TCP peer that is initiating communication includes them in their synchronize (SYN) segment.

See the references for more information.

Affected Software/OS
TCP/IPv4 implementations that implement RFC1323.

Vulnerability Insight
The remote host implements TCP timestamps, as defined by RFC1323.

Vulnerability Detection Method
Special IP packets are forged and sent with a little delay in between to the target IP. The responses are searched for a timestamps. If found, the timestamps are reported.

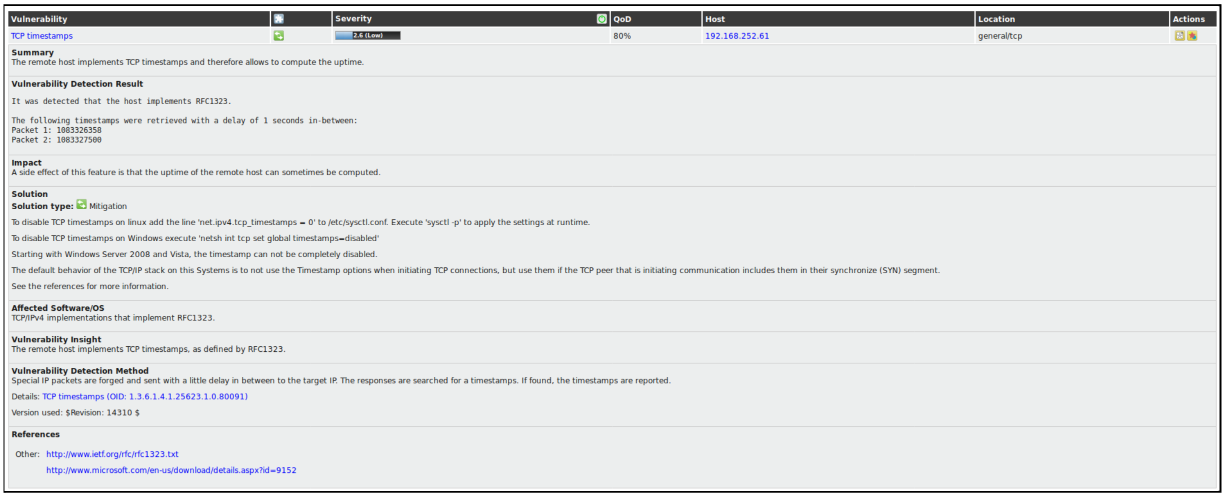
Details: TCP timestamps (OID: 1.3.6.1.4.1.25623.1.0.80091)

Version used: $Revision: 14310 $

References

Other: http://www.ietf.org/rfc/rfc1323.txt
http://www.microsoft.com/en-us/download/details.aspx?id=9152

Figure 1.7: Overview of ‘TCP Timestamp Vulnerability’ on 192.168.252.3 machine



Vulnerability		Severity	QoD	Host	Location	Actions
TCP timestamps		2.6 (Low)	80%	192.168.252.61	general/tcp	

Summary
The remote host implements TCP timestamps and therefore allows to compute the uptime.

Vulnerability Detection Result

```
It was detected that the host implements RFC1323.

The following timestamps were retrieved with a delay of 1 seconds in-between:
Packet 1: 1083326358
Packet 2: 1083327500
```

Impact
A side effect of this feature is that the uptime of the remote host can sometimes be computed.

Solution
Solution type: Mitigation

To disable TCP timestamps on linux add the line 'net.ipv4.tcp_timestamps = 0' to /etc/sysctl.conf. Execute 'sysctl -p' to apply the settings at runtime.

To disable TCP timestamps on Windows execute 'netsh int tcp set global timestamps=disabled'

Starting with Windows Server 2008 and Vista, the timestamp can not be completely disabled.

The default behavior of the TCP/IP stack on this Systems is to not use the Timestamp options when initiating TCP connections, but use them if the TCP peer that is initiating communication includes them in their synchronize (SYN) segment.

See the references for more information.

Affected Software/OS
TCP/IPv4 implementations that implement RFC1323.

Vulnerability Insight
The remote host implements TCP timestamps, as defined by RFC1323.

Vulnerability Detection Method
Special IP packets are forged and sent with a little delay in between to the target IP. The responses are searched for a timestamps. If found, the timestamps are reported.

Details: TCP timestamps (OID: 1.3.6.1.4.1.25623.1.0.80091)

Version used: $Revision: 14310 $

References

Other: http://www.ietf.org/rfc/rfc1323.txt
http://www.microsoft.com/en-us/download/details.aspx?id=9152

Figure 1.8: Overview of ‘TCP Timestamp Vulnerability’ on 192.168.252.61 machine

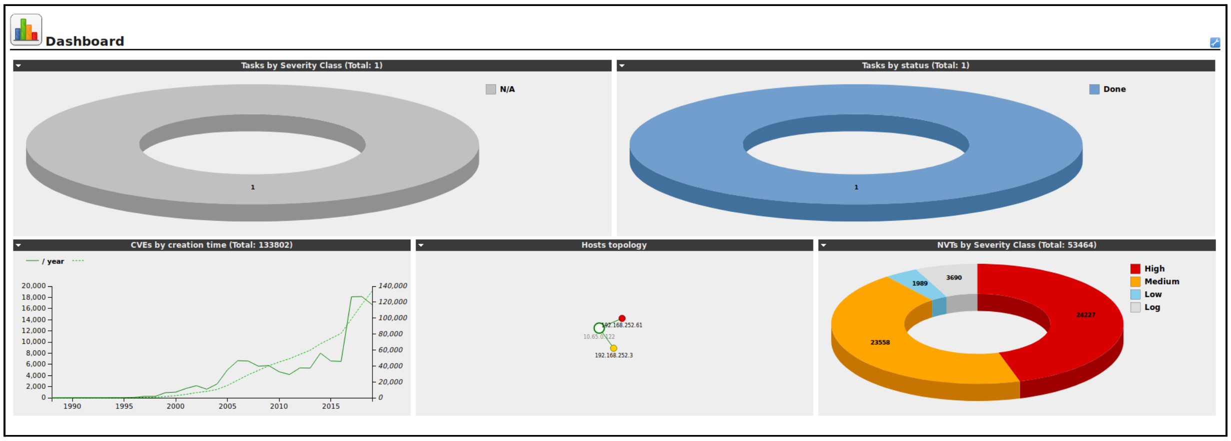


Figure 1.9: Dashboard generated by the results