# Scan Report

October 9, 2018

**Summary**

This document reports on the results of an automatic security scan. All dates are displayed using the timezone "Coordinated Universal Time", which is abbreviated "UTC". The task was "Example Scans". The scan started at Tue Feb 21 15:24:31 2017 UTC and ended at Tue Feb 21 18:11:04 2017 UTC. The report first summarises the results found. Then, for each host, the report describes every issue found. Please consider the advice given in each description, in order to rectify the issue.

## Contents

**1 Result Overview** **2**

1.1 Host Authentications . . . 3

**2 Results per Host** **3**

2.1 127.0.0.46 . . . 3

2.1.1 High 445/tcp . . . 3

2.1.2 High 22/tcp . . . 6

2.1.3 Medium 135/tcp . . . 8

2.1.4 Medium 22/tcp . . . 9

2.1.5 Low 22/tcp . . . 10

2.2 127.0.0.4 . . . 10

2.2.1 High 22/tcp . . . 11

2.2.2 High 445/tcp . . . 11

2.2.3 Medium 22/tcp . . . 12

2.2.4 Medium 80/tcp . . . 13

2.2.5 Medium 135/tcp . . . 14

2.2.6 Medium 3389/tcp . . . 14

2.2.7 Low 22/tcp . . . 16

2.2.8 Low general/tcp . . . 17

2.3 127.0.0.23 . . . 18

2.3.1 High 445/tcp . . . 18

2.3.2 Medium 3389/tcp . . . 21

2.3.3 Medium 135/tcp . . . 23
2.3.4 Low general/tcp . . . 23
2.4 127.0.0.29 . . . 24
2.4.1 High 3389/tcp . . . 24
2.4.2 High 445/tcp . . . 25
2.5 127.0.0.14 . . . 28
2.5.1 High 445/tcp . . . 29
2.5.2 Medium 389/tcp . . . 29
2.5.3 Medium 80/tcp . . . 30
2.5.4 Medium 3389/tcp . . . 31
2.5.5 Medium 3269/tcp . . . 34
2.5.6 Medium 3268/tcp . . . 38
2.5.7 Medium 135/tcp . . . 38
2.5.8 Medium 443/tcp . . . 39
2.5.9 Medium 636/tcp . . . 44
2.5.10 Low general/tcp . . . 48
2.6 127.0.0.1 . . . 49
2.6.1 High 445/tcp . . . 50
2.6.2 High 22/tcp . . . 51
2.6.3 Medium 135/tcp . . . 54
2.6.4 Medium 22/tcp . . . 54
2.6.5 Medium 3389/tcp . . . 55
2.6.6 Low 22/tcp . . . 57
2.6.7 Low general/tcp . . . 57
2.7 127.0.0.10 . . . 58
2.7.1 High 445/tcp . . . 59
2.7.2 High 22/tcp . . . 62
2.7.3 Medium 3389/tcp . . . 64
2.7.4 Medium 22/tcp . . . 65
2.7.5 Medium 135/tcp . . . 66
2.7.6 Low general/tcp . . . 67
2.7.7 Low 22/tcp . . . 68
2.8 127.0.0.22 . . . 68
2.8.1 High 3389/tcp . . . 69
2.8.2 High 445/tcp . . . 70
2.8.3 Medium 135/tcp . . . 72
2.9 127.0.0.44 . . . 73
2.9.1 High 22/tcp . . . 73
2.9.2 Medium 22/tcp . . . 74
2.9.3 Low general/tcp . . . 75

2.9.4 Low 22/tcp . . . 76
2.10 127.0.0.26 . . . 76
2.10.1 High 445/tcp . . . 76
2.10.2 High 3389/tcp . . . 79
2.10.3 Medium 135/tcp . . . 80
2.10.4 Medium 22/tcp . . . 81
2.10.5 Low 22/tcp . . . 82
2.11 127.0.0.13 . . . 82
2.11.1 High 22/tcp . . . 82
2.11.2 High general/tcp . . . 84
2.11.3 High 445/tcp . . . 85
2.11.4 Medium 22/tcp . . . 86
2.11.5 Medium 2011/tcp . . . 87
2.11.6 Low 22/tcp . . . 92
2.12 127.0.0.7 . . . 92
2.12.1 High 3389/tcp . . . 92
2.12.2 High 445/tcp . . . 93
2.12.3 Medium 135/tcp . . . 96
2.12.4 Medium 8080/tcp . . . 97
2.12.5 Medium 443/tcp . . . 97
2.12.6 Medium 8098/tcp . . . 99
2.12.7 Medium 80/tcp . . . 105
2.12.8 Medium 21/tcp . . . 106
2.13 127.0.0.20 . . . 107
2.13.1 High 445/tcp . . . 108
2.13.2 Medium 3389/tcp . . . 108
2.13.3 Medium 135/tcp . . . 110
2.13.4 Medium 8080/tcp . . . 111
2.13.5 Medium 443/tcp . . . 112
2.13.6 Low general/tcp . . . 115
2.14 127.0.0.31 . . . 116
2.14.1 High 445/tcp . . . 117
2.14.2 Medium 135/tcp . . . 117
2.14.3 Medium 3389/tcp . . . 118
2.14.4 Low general/tcp . . . 121
2.15 127.0.0.34 . . . 122
2.15.1 High 445/tcp . . . 122
2.15.2 Medium 636/tcp . . . 123
2.15.3 Medium 389/tcp . . . 127
2.15.4 Medium 3389/tcp . . . 127

2.15.5 Medium 135/tcp . . . 129
2.15.6 Low general/tcp . . . 130
2.16 127.0.0.25 . . . 131
2.16.1 High 445/tcp . . . 131
2.16.2 Medium 135/tcp . . . 134
2.16.3 Medium 3389/tcp . . . 135
2.16.4 Low general/tcp . . . 137
2.17 127.0.0.36 . . . 138
2.17.1 High 445/tcp . . . 138
2.17.2 Medium 135/tcp . . . 139
2.17.3 Low general/tcp . . . 139
2.18 127.0.0.47 . . . 140
2.18.1 High 445/tcp . . . 140
2.18.2 Low general/tcp . . . 141
2.19 127.0.0.8 . . . 142
2.19.1 High 445/tcp . . . 142
2.19.2 High 636/tcp . . . 143
2.19.3 Medium 135/tcp . . . 144
2.19.4 Medium 3268/tcp . . . 145
2.19.5 Medium 443/tcp . . . 145
2.19.6 Medium 3389/tcp . . . 150
2.19.7 Medium 389/tcp . . . 152
2.19.8 Medium 3269/tcp . . . 153
2.19.9 Medium 636/tcp . . . 157
2.19.10 Low general/tcp . . . 161
2.20 127.0.0.35 . . . 162
2.20.1 High 445/tcp . . . 162
2.20.2 High general/tcp . . . 163
2.21 127.0.0.39 . . . 163
2.21.1 High 445/tcp . . . 164
2.21.2 Medium 135/tcp . . . 164
2.21.3 Medium 3389/tcp . . . 165
2.21.4 Low general/tcp . . . 166
2.22 127.0.0.2 . . . 167
2.22.1 High 445/tcp . . . 167
2.22.2 Medium 3389/tcp . . . 168
2.22.3 Medium 135/tcp . . . 171
2.22.4 Low general/tcp . . . 171
2.23 127.0.0.6 . . . 172
2.23.1 High 22/tcp . . . 173

2.23.2 Medium 9390/tcp . . . 173
2.23.3 Medium 22/tcp . . . 175
2.23.4 Medium 443/tcp . . . 176
2.23.5 Low general/tcp . . . 177
2.23.6 Low 22/tcp . . . 178
2.24 127.0.0.3 . . . 179
2.24.1 High 22/tcp . . . 179
2.24.2 Medium 443/tcp . . . 179
2.24.3 Medium 9390/tcp . . . 180
2.24.4 Low general/tcp . . . 181
2.25 127.0.0.43 . . . 182
2.25.1 High 22/tcp . . . 183
2.26 127.0.0.28 . . . 183
2.26.1 High 22/tcp . . . 183
2.26.2 Low general/tcp . . . 184
2.27 127.0.0.32 . . . 185
2.27.1 High 22/tcp . . . 185
2.27.2 Low general/tcp . . . 186
2.28 127.0.0.5 . . . 187
2.28.1 High 901/tcp . . . 187
2.28.2 Medium 22/tcp . . . 188
2.28.3 Low 22/tcp . . . 189
2.28.4 Low general/tcp . . . 189
2.29 127.0.0.38 . . . 190
2.29.1 High 901/tcp . . . 190
2.29.2 Medium 22/tcp . . . 191
2.29.3 Low general/tcp . . . 192
2.29.4 Low 22/tcp . . . 193
2.30 127.0.0.41 . . . 193
2.30.1 Medium 443/tcp . . . 193
2.30.2 Medium 22/tcp . . . 197
2.31 127.0.0.27 . . . 198
2.31.1 Medium 22/tcp . . . 198
2.31.2 Medium 3871/tcp . . . 199
2.31.3 Low 22/tcp . . . 203
2.31.4 Low general/tcp . . . 203
2.32 127.0.0.15 . . . 204
2.32.1 Medium 22/tcp . . . 204
2.32.2 Medium 443/tcp . . . 205
2.32.3 Low general/tcp . . . 208

2.32.4 Low 22/tcp . . . 209
2.33 127.0.0.17 . . . 209
2.33.1 Medium 135/tcp . . . 210
2.33.2 Medium 3389/tcp . . . 210
2.33.3 Low general/tcp . . . 212
2.34 127.0.0.19 . . . 213
2.34.1 Medium 80/tcp . . . 213
2.34.2 Medium 22/tcp . . . 214
2.34.3 Low general/tcp . . . 215
2.34.4 Low 22/tcp . . . 216
2.35 127.0.0.12 . . . 216
2.35.1 Medium 22/tcp . . . 217
2.35.2 Low general/tcp . . . 217
2.35.3 Low 22/tcp . . . 218
2.36 127.0.0.42 . . . 219
2.36.1 Medium 22/tcp . . . 219
2.36.2 Medium 443/tcp . . . 220
2.36.3 Low 22/tcp . . . 224
2.37 127.0.0.40 . . . 224
2.37.1 Medium 22/tcp . . . 224
2.37.2 Low general/tcp . . . 225
2.37.3 Low 22/tcp . . . 226
2.38 127.0.0.11 . . . 227
2.38.1 Medium 22/tcp . . . 227
2.38.2 Low general/tcp . . . 228
2.38.3 Low 22/tcp . . . 229
2.39 127.0.0.37 . . . 229
2.39.1 Medium 80/tcp . . . 229
2.39.2 Medium 22/tcp . . . 230
2.39.3 Low 22/tcp . . . 231
2.39.4 Low general/tcp . . . 232
2.40 127.0.0.21 . . . 233
2.40.1 Medium 22/tcp . . . 233
2.40.2 Low general/tcp . . . 234
2.40.3 Low 22/tcp . . . 235
2.41 127.0.0.16 . . . 235
2.41.1 Medium 22/tcp . . . 235
2.41.2 Low 22/tcp . . . 236
2.41.3 Low general/tcp . . . 237
2.42 127.0.0.24 . . . 238

2.42.1 Medium 443/tcp . . . . . 238
2.43 127.0.0.45 . . . . . 240
2.43.1 Low general/tcp . . . . . 240
2.44 127.0.0.30 . . . . . 242
2.44.1 Low general/tcp . . . . . 242
2.45 127.0.0.48 . . . . . 243
2.45.1 Low general/tcp . . . . . 243
2.46 127.0.0.18 . . . . . 244
2.46.1 Low general/tcp . . . . . 244

# 1 Result Overview

| Host | High | Medium | Low | Log | False Positive |
|---|---|---|---|---|---|
| 127.0.0.46 | 5 | 2 | 1 | 0 | 0 |
| 127.0.0.4 | 2 | 5 | 2 | 0 | 0 |
| 127.0.0.23 | 3 | 3 | 1 | 0 | 0 |
| 127.0.0.29 | 4 | 0 | 0 | 0 | 0 |
| 127.0.0.14 | 1 | 23 | 1 | 0 | 0 |
| 127.0.0.1 | 4 | 4 | 2 | 0 | 0 |
| 127.0.0.10 | 5 | 4 | 2 | 0 | 0 |
| 127.0.0.22 | 4 | 1 | 0 | 0 | 0 |
| 127.0.0.44 | 1 | 1 | 2 | 0 | 0 |
| 127.0.0.26 | 4 | 2 | 1 | 0 | 0 |
| 127.0.0.13 | 4 | 7 | 1 | 0 | 0 |
| 127.0.0.7 | 4 | 14 | 0 | 0 | 0 |
| 127.0.0.20 | 1 | 8 | 1 | 0 | 0 |
| 127.0.0.31 | 1 | 4 | 1 | 0 | 0 |
| 127.0.0.34 | 1 | 9 | 1 | 0 | 0 |
| 127.0.0.25 | 3 | 3 | 1 | 0 | 0 |
| 127.0.0.36 | 1 | 1 | 1 | 0 | 0 |
| 127.0.0.47 | 1 | 0 | 1 | 0 | 0 |
| 127.0.0.8 | 2 | 20 | 1 | 0 | 0 |
| 127.0.0.35 | 2 | 0 | 0 | 0 | 0 |
| 127.0.0.39 | 1 | 2 | 1 | 0 | 0 |
| 127.0.0.2 | 1 | 4 | 1 | 0 | 0 |
| 127.0.0.6 | 1 | 5 | 2 | 0 | 0 |
| 127.0.0.3 | 1 | 2 | 1 | 0 | 0 |
| 127.0.0.43 | 1 | 0 | 0 | 0 | 0 |
| 127.0.0.28 | 1 | 0 | 1 | 0 | 0 |
| 127.0.0.32 | 1 | 0 | 1 | 0 | 0 |
| 127.0.0.5 | 1 | 1 | 2 | 0 | 0 |
| 127.0.0.38 | 1 | 1 | 2 | 0 | 0 |
| 127.0.0.41 | 0 | 5 | 0 | 0 | 0 |
| 127.0.0.27 | 0 | 5 | 2 | 0 | 0 |
| 127.0.0.15 | 0 | 4 | 2 | 0 | 0 |
| 127.0.0.17 | 0 | 3 | 1 | 0 | 0 |
| 127.0.0.19 | 0 | 2 | 2 | 0 | 0 |
| 127.0.0.12 | 0 | 1 | 2 | 0 | 0 |
| 127.0.0.42 | 0 | 5 | 1 | 0 | 0 |
| 127.0.0.40 | 0 | 1 | 2 | 0 | 0 |
| 127.0.0.11 | 0 | 1 | 2 | 0 | 0 |
| 127.0.0.37 | 0 | 2 | 2 | 0 | 0 |
| 127.0.0.21 | 0 | 1 | 2 | 0 | 0 |
| 127.0.0.16 | 0 | 1 | 2 | 0 | 0 |
| 127.0.0.24 | 0 | 2 | 0 | 0 | 0 |
| 127.0.0.45 | 0 | 0 | 2 | 0 | 0 |

. . . (continues) . . .

... (continued) ...

| Host | High | Medium | Low | Log | False Positive |
|---|---|---|---|---|---|
| 127.0.0.30 | 0 | 0 | 1 | 0 | 0 |
| 127.0.0.48 | 0 | 0 | 1 | 0 | 0 |
| 127.0.0.18 | 0 | 0 | 1 | 0 | 0 |
| Total: 46 | 62 | 159 | 56 | 0 | 0 |

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.
Notes are included in the report.
This report might not show details of all issues that were found.
It only lists hosts that produced issues.
Issues with the threat level "Log" are not shown.
Issues with the threat level "Debug" are not shown.
Issues with the threat level "False Positive" are not shown.
Only results with a minimum QoD of 70 are shown.

This report contains all 277 results selected by the filtering described above. Before filtering there were 278 results.

## 1.1 Host Authentications

| Host | Protocol | Result | Port/User |
|---|---|---|---|
| 127.0.0.1 | SMB | Success | Protocol SMB, Port 445, User |
| 127.0.0.5 | SMB | Success | Protocol SMB, Port 445, User |
| 127.0.0.38 | SMB | Success | Protocol SMB, Port 445, User |
| 127.0.0.45 | ESXi | Failure | Protocol ESXi, Port 443, User : Login failure |

# 2 Results per Host

## 2.1 127.0.0.46

Host scan start Tue Feb 21 15:24:48 2017 UTC
Host scan end Tue Feb 21 16:32:45 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 445/tcp | High |
| 22/tcp | High |
| 135/tcp | Medium |
| 22/tcp | Medium |
| 22/tcp | Low |

### 2.1.1 High 445/tcp

High (CVSS: 10.0)
NVT: Vulnerabilities in SMB Could Allow Remote Code Execution (958687) - Remote

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS09-001.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation could allow remote unauthenticated attackers to cause denying the service by sending a specially crafted network message to a system running the server service.
Impact Level: System/Network

**Solution**
**Solution type:** VendorFix
Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory from the below link, http://www.microsoft.com/technet/security/bulletin/ms09-001.mspx

**Affected Software/OS**
Microsoft Windows 2K Service Pack 4 and prior.
Microsoft Windows XP Service Pack 3 and prior.
Microsoft Windows 2003 Service Pack 2 and prior.

**Vulnerability Insight**
The issue is due to the way Server Message Block (SMB) Protocol software handles specially crafted SMB packets.

**Vulnerability Detection Method**
Details: `Vulnerabilities in SMB Could Allow Remote Code Execution (958687) - Remote`
OID:1.3.6.1.4.1.25623.1.0.900233
Version used: `$Revision: 4692 $`

**References**
```
CVE: CVE-2008-4114, CVE-2008-4834, CVE-2008-4835
BID:31179
Other:
  URL:http://www.milw0rm.com/exploits/6463
   URL:http://www.microsoft.com/technet/security/bulletin/ms09-001.mspx
```

High (CVSS: 0.0)
NVT: SMBv1 enabled (Remote Check)

**Summary**
The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.

. . . continues on next page . . .

... continued from previous page ...

This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices.

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

**Log Method**
Details: `SMBv1 enabled (Remote Check)`
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: `$Revision: 5222 $`

**References**
```
Other:
  URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best
↪-Practices
   URL:https://support.microsoft.com/en-us/kb/2696547
   URL:https://support.microsoft.com/en-us/kb/204279
   URL:https://technet.microsoft.com/en-us/library/security/MS17-010
```

High (CVSS: 10.0)
NVT: Microsoft Windows SMB Server NTLM Multiple Vulnerabilities (971468)

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS10-012.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow remote attackers to execute arbitrary code or cause a denial of service or bypass the authentication mechanism via brute force technique. Impact Level: System/Application

**Solution**
**Solution type:** VendorFix

... continues on next page ...

... continued from previous page ...

Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory from the below link, http://www.microsoft.com/technet/security/bulletin/ms10-012.mspx

**Affected Software/OS**
Microsoft Windows 7 Microsoft Windows 2000 Service Pack and prior Microsoft Windows XP Service Pack 3 and prior Microsoft Windows Vista Service Pack 2 and prior Microsoft Windows Server 2003 Service Pack 2 and prior Microsoft Windows Server 2008 Service Pack 2 and prior

**Vulnerability Insight**
- An input validation error exists while processing SMB requests and can be exploited to cause a buffer overflow via a specially crafted SMB packet. - An error exists in the SMB implementation while parsing SMB packets during the Negotiate phase causing memory corruption via a specially crafted SMB packet. - NULL pointer dereference error exists in SMB while verifying the 'share' and 'servername' fields in SMB packets causing denial of service. - A lack of cryptographic entropy when the SMB server generates challenges during SMB NTLM authentication and can be exploited to bypass the authentication mechanism.

**Vulnerability Detection Method**
Details: `Microsoft Windows SMB Server NTLM Multiple Vulnerabilities (971468)`
OID:1.3.6.1.4.1.25623.1.0.902269
Version used: `$Revision: 5136 $`

**References**
```
CVE: CVE-2010-0020, CVE-2010-0021, CVE-2010-0022, CVE-2010-0231
Other:
  URL:http://secunia.com/advisories/38510/
   URL:http://support.microsoft.com/kb/971468
   URL:http://www.vupen.com/english/advisories/2010/0345
   URL:http://www.microsoft.com/technet/security/bulletin/ms10-012.mspx
```

[ return to 127.0.0.46 ]

### 2.1.2 High 22/tcp

High (CVSS: 7.8)
NVT: OpenSSH Denial of Service And User Enumeration Vulnerabilities (Windows)

**Summary**
This host is installed with openssh and is prone to denial of service and user enumeration vulnerabilities.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**

... continues on next page ...

... continued from previous page ...

Successfully exploiting this issue allows remote attackers to cause a denial of service (crypt CPU consumption) and to enumerate users by leveraging the timing difference between responses when a large password is provided.

**Solution**
**Solution type:** VendorFix
Upgrade to OpenSSH version 7.3 or later. For updates refer to http://www.openssh.com

**Affected Software/OS**
OpenSSH versions before 7.3 on Windows

**Vulnerability Insight**
Multiple flaws exists due to,
- The auth_password function in 'auth-passwd.c' script does not limit password lengths for password authentication.
- The sshd in OpenSSH, when SHA256 or SHA512 are used for user password hashing uses BLOWFISH hashing on a static password when the username does not exist and it takes much longer to calculate SHA256/SHA512 hash than BLOWFISH hash.

**Vulnerability Detection Method**
Checks if a vulnerable version is present on the target host.
Details: `OpenSSH Denial of Service And User Enumeration Vulnerabilities (Windows)`
OID:1.3.6.1.4.1.25623.1.0.809121
Version used: `$Revision: 5083 $`

**References**
`CVE: CVE-2016-6515, CVE-2016-6210`
`BID:92212`
`Other:`
`URL:http://www.openssh.com/txt/release-7.3`
`URL:http://seclists.org/fulldisclosure/2016/Jul/51`
`URL:https://security-tracker.debian.org/tracker/CVE-2016-6210`
`URL:http://openwall.com/lists/oss-security/2016/08/01/2`

High (CVSS: 7.5)
NVT: OpenSSH Multiple Vulnerabilities Jan17 (Windows)

**Summary**
This host is installed with openssh and is prone to multiple vulnerabilities.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successfully exploiting this issue allows local users to obtain sensitive private-key information, to gain privileges, conduct a senial-of-service condition and allows remote attackers to execute arbitrary local PKCS#11 modules.

... continues on next page ...

... continued from previous page ...

Impact Level: Application

**Solution**
**Solution type:** VendorFix
Upgrade to OpenSSH version 7.4 or later. For updates refer to http://www.openssh.com

**Affected Software/OS**
OpenSSH versions before 7.4 on Windows

**Vulnerability Insight**
Multiple flaws exists due to, - An 'authfile.c' script does not properly consider the effects of realloc on buffer contents. - The shared memory manager (associated with pre-authentication compression) does not ensure that a bounds check is enforced by all compilers. - The sshd in OpenSSH creates forwarded Unix-domain sockets as root, when privilege separation is not used. - An untrusted search path vulnerability in ssh-agent.c in ssh-agent. - NULL pointer dereference error due to an out-of-sequence NEWKEYS message.

**Vulnerability Detection Method**
Get the installed version with the help of detect NVT and check the version is vulnerable or not.
Details: `OpenSSH Multiple Vulnerabilities Jan17 (Windows)`
OID:1.3.6.1.4.1.25623.1.0.810325
Version used: `$Revision: 5084 $`

**References**
```
CVE: CVE-2016-10009, CVE-2016-10010, CVE-2016-10011, CVE-2016-10012, CVE-2016-10
↪708
BID:94968, 94972, 94977, 94975
Other:
  URL:https://www.openssh.com/txt/release-7.4
   URL:http://www.openwall.com/lists/oss-security/2016/12/19/2
   URL:http://blog.swiecki.net/2018/01/fuzzing-tcp-servers.html
   URL:https://anongit.mindrot.org/openssh.git/commit/?id=28652bca29046f62c7045e
↪933e6b931de1d16737
```

[ return to 127.0.0.46 ]

### 2.1.3 Medium 135/tcp

Medium (CVSS: 5.0)
NVT: DCE/RPC and MSRPC Services Enumeration Reporting

**Summary**
Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries.

... continues on next page ...

... continued from previous page ...

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker may use this fact to gain more knowledge about the remote host.

**Solution**
**Solution type:** Mitigation
Filter incoming traffic to this ports.

**Vulnerability Detection Method**
Details: `DCE/RPC and MSRPC Services Enumeration Reporting`
OID:1.3.6.1.4.1.25623.1.0.10736
Version used: `$Revision: 4998 $`

[ return to 127.0.0.46 ]

### 2.1.4 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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

... continues on next page ...

... continued from previous page ...

Version used: $Revision: 4490 $

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

[ return to 127.0.0.46 ]

### 2.1.5 Low 22/tcp

Low (CVSS: 2.6)
NVT: SSH Weak MAC Algorithms Supported

**Summary**
The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Disable the weak MAC algorithms.

**Vulnerability Detection Method**
Details: SSH Weak MAC Algorithms Supported
OID:1.3.6.1.4.1.25623.1.0.105610
Version used: $Revision: 4490 $

[ return to 127.0.0.46 ]

## 2.2 127.0.0.4

Host scan start Tue Feb 21 15:24:50 2017 UTC
Host scan end Tue Feb 21 15:55:39 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 22/tcp | High |
| 445/tcp | High |
| 22/tcp | Medium |
| 80/tcp | Medium |
| 135/tcp | Medium |
| 3389/tcp | Medium |
| 22/tcp | Low |
| general/tcp | Low |

### 2.2.1 High 22/tcp

| High (CVSS: 7.5)<br>NVT: SSH Brute Force Logins With Default Credentials Reporting |
|---|
| **Summary**<br>It was possible to login into the remote SSH server using default credentials.<br>As the NVT 'SSH Brute Force Logins with default Credentials' (OID: 1.3.6.1.4.1.25623.1.0.108013) might run into a timeout the actual reporting of this vulnerability takes place in this NVT instead. The script preference 'Report timeout' allows you to configure if such an timeout is reported. |
| **Vulnerability Detection Result**<br>Vulnerability was detected according to the Vulnerability Detection Method. |
| **Solution**<br>**Solution type:** Mitigation<br>Change the password as soon as possible. |
| **Vulnerability Detection Method**<br>Try to login with a number of known default credentials via the SSH protocol.<br>Details: `SSH Brute Force Logins With Default Credentials Reporting`<br>OID:1.3.6.1.4.1.25623.1.0.103239<br>Version used: `$Revision: 4508 $` |

[ return to 127.0.0.4 ]

### 2.2.2 High 445/tcp

| High (CVSS: 0.0)<br>NVT: SMBv1 enabled (Remote Check) |
|---|
| **Summary**<br>The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.<br>This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810). |
| **Vulnerability Detection Result**<br>Vulnerability was detected according to the Vulnerability Detection Method. |
| **Solution**<br>**Solution type:** Workaround<br>Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices. |
| |
| . . . continues on next page . . . |

... continued from previous page ...

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

**Log Method**
Details: `SMBv1 enabled (Remote Check)`
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: `$Revision: 5222 $`

**References**
```
Other:
  URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best
↪-Practices
   URL:https://support.microsoft.com/en-us/kb/2696547
   URL:https://support.microsoft.com/en-us/kb/204279
   URL:https://technet.microsoft.com/en-us/library/security/MS17-010
```

[ return to 127.0.0.4 ]

### 2.2.3 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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`

... continues on next page ...

... continued from previous page ...

OID:1.3.6.1.4.1.25623.1.0.105611
Version used: $Revision: 4490 $

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

[ return to 127.0.0.4 ]

### 2.2.4 Medium 80/tcp

Medium (CVSS: 5.0)
NVT: Microsoft IIS Default Welcome Page Information Disclosure Vulnerability

**Summary**
The host is running Microsoft IIS Webserver and is prone to information disclosure vulnerability.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow remote attackers to obtain sensitive information that could aid in further attacks.

**Solution**
**Solution type:** Mitigation
Disable the default pages within the server configuration.

**Affected Software/OS**
Microsoft Internet Information Services

**Vulnerability Insight**
The flaw is due to misconfiguration of IIS Server, which allows to access default pages when the server is not used.

**Vulnerability Detection Method**
Details: Microsoft IIS Default Welcome Page Information Disclosure Vulnerability
OID:1.3.6.1.4.1.25623.1.0.802806
Version used: $Revision: 2715 $

**References**
Other:
URL:http://www.iis.net/

[ return to 127.0.0.4 ]

### 2.2.5 Medium 135/tcp

| Medium (CVSS: 5.0)<br>NVT: DCE/RPC and MSRPC Services Enumeration Reporting |
|---|
| **Summary**<br>Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries. |
| **Vulnerability Detection Result**<br>Vulnerability was detected according to the Vulnerability Detection Method. |
| **Impact**<br>An attacker may use this fact to gain more knowledge about the remote host. |
| **Solution**<br>**Solution type:** Mitigation<br>Filter incoming traffic to this ports. |
| **Vulnerability Detection Method**<br>Details: `DCE/RPC and MSRPC Services Enumeration Reporting`<br>OID:1.3.6.1.4.1.25623.1.0.10736<br>Version used: `$Revision: 4998 $` |

[ return to 127.0.0.4 ]

### 2.2.6 Medium 3389/tcp

| Medium (CVSS: 4.3)<br>NVT: SSL/TLS: Report Weak Cipher Suites |
|---|
| **Summary**<br>This routine reports all Weak SSL/TLS cipher suites accepted by a service.<br>NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication. |
| **Vulnerability Detection Result**<br>Vulnerability was detected according to the Vulnerability Detection Method. |
| **Solution**<br>**Solution type:** Mitigation<br>The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.<br>Please see the references for more resources supporting you with this task. |
| . . . continues on next page . . . |

... continued from previous page ...

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Weak Cipher Suites`
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: `$Revision: 4863 $`

**References**
`CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000`
`Other:`
`URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html`
`URL:https://bettercrypto.org/`
`URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/`

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)

... continues on next page ...

... continued from previous page ...

Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: `SSL/TLS: Certificate Signed Using A Weak Signature Algorithm`
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: `$Revision: 4781 $`

**References**
```
Other:
  URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with
↪-sha-1-based-signature-algorithms/
```

[ return to 127.0.0.4 ]

### 2.2.7 Low 22/tcp

Low (CVSS: 2.6)
NVT: SSH Weak MAC Algorithms Supported

**Summary**
The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Disable the weak MAC algorithms.

**Vulnerability Detection Method**
Details: `SSH Weak MAC Algorithms Supported`
OID:1.3.6.1.4.1.25623.1.0.105610
Version used: `$Revision: 4490 $`

[ return to 127.0.0.4 ]

### 2.2.8 Low general/tcp

| Low (CVSS: 2.6)<br>NVT: TCP timestamps |
|---|
| **Summary**<br>The remote host implements TCP timestamps and therefore allows to compute the uptime. |
| **Vulnerability Detection Result**<br>Vulnerability was detected according to the Vulnerability Detection Method. |
| **Impact**<br>A side effect of this feature is that the uptime of the remote host can sometimes be computed. |
| **Solution**<br>**Solution type:** Mitigation<br>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.<br>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.<br>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.<br>See also: http://www.microsoft.com/en-us/download/details.aspx?id=9152 |
| **Affected Software/OS**<br>TCP/IPv4 implementations that implement RFC1323. |
| **Vulnerability Insight**<br>The remote host implements TCP timestamps, as defined by RFC1323. |
| **Vulnerability Detection Method**<br>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.<br>Details: `TCP timestamps`<br>OID:1.3.6.1.4.1.25623.1.0.80091<br>Version used: `$Revision: 5309 $` |
| **References**<br>`Other:`<br>`URL:http://www.ietf.org/rfc/rfc1323.txt` |

[ return to 127.0.0.4 ]

## 2.3 127.0.0.23

Host scan start Tue Feb 21 15:24:50 2017 UTC
Host scan end Tue Feb 21 16:27:05 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 445/tcp | High |
| 3389/tcp | Medium |
| 135/tcp | Medium |
| general/tcp | Low |

### 2.3.1 High 445/tcp

High (CVSS: 10.0)
NVT: Microsoft Windows SMB2 Negotiation Protocol Remote Code Execution Vulnerability

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS09-050.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker can exploit this issue to execute code with SYSTEM-level privileges failed exploit attempts will likely cause denial-of-service conditions.
Impact Level: System

**Solution**
**Solution type:** VendorFix

**Affected Software/OS**
- Windows 7 RC
- Windows Vista and
- Windows 2008 Server

**Vulnerability Insight**
Multiple vulnerabilities exists,
- A denial of service vulnerability exists in the way that Microsoft Server Message Block (SMB) Protocol software handles specially crafted SMB version 2 (SMBv2) packets.
- Unauthenticated remote code execution vulnerability exists in the way that Microsoft Server Message Block (SMB) Protocol software handles specially crafted SMB packets.

**Vulnerability Detection Method**
Details: `Microsoft Windows SMB2 Negotiation Protocol Remote Code Execution Vulnerability`
OID:1.3.6.1.4.1.25623.1.0.900965
Version used: `$Revision: 5074 $`

. . . continues on next page . . .

... continued from previous page ...

**References**
CVE: CVE-2009-2526, CVE-2009-2532, CVE-2009-3103
BID:36299
Other:
URL:http://www.microsoft.com/technet/security/bulletin/MS09-050.mspx

**Note**

This is a sample note on this scan result which I would like to see for any othe
↪r occurance of
this vulnerability, regardless of the task or host.

Last modified: Thu Mar 23 16:52:39 2017 UTC

High (CVSS: 0.0)
NVT: SMBv1 enabled (Remote Check)

**Summary**
The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.
This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices.

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

**Log Method**
Details: SMBv1 enabled (Remote Check)
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: $Revision: 5222 $

**References**
Other:
URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best
↪-Practices
URL:https://support.microsoft.com/en-us/kb/2696547

... continues on next page ...

... continued from previous page ...

URL:https://support.microsoft.com/en-us/kb/204279
URL:https://technet.microsoft.com/en-us/library/security/MS17-010

High (CVSS: 10.0)
NVT: Microsoft Windows SMB Server NTLM Multiple Vulnerabilities (971468)

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS10-012.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow remote attackers to execute arbitrary code or cause a denial of service or bypass the authentication mechanism via brute force technique. Impact Level: System/Application

**Solution**
**Solution type:** VendorFix
Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory from the below link, http://www.microsoft.com/technet/security/bulletin/ms10-012.mspx

**Affected Software/OS**
Microsoft Windows 7 Microsoft Windows 2000 Service Pack and prior Microsoft Windows XP Service Pack 3 and prior Microsoft Windows Vista Service Pack 2 and prior Microsoft Windows Server 2003 Service Pack 2 and prior Microsoft Windows Server 2008 Service Pack 2 and prior

**Vulnerability Insight**
- An input validation error exists while processing SMB requests and can be exploited to cause a buffer overflow via a specially crafted SMB packet. - An error exists in the SMB implementation while parsing SMB packets during the Negotiate phase causing memory corruption via a specially crafted SMB packet. - NULL pointer dereference error exists in SMB while verifying the 'share' and 'servername' fields in SMB packets causing denial of service. - A lack of cryptographic entropy when the SMB server generates challenges during SMB NTLM authentication and can be exploited to bypass the authentication mechanism.

**Vulnerability Detection Method**
Details: Microsoft Windows SMB Server NTLM Multiple Vulnerabilities (971468)
OID:1.3.6.1.4.1.25623.1.0.902269
Version used: $Revision: 5136 $

**References**
CVE: CVE-2010-0020, CVE-2010-0021, CVE-2010-0022, CVE-2010-0231
Other:
URL:http://secunia.com/advisories/38510/

... continues on next page ...

... continued from previous page ...

URL:http://support.microsoft.com/kb/971468
URL:http://www.vupen.com/english/advisories/2010/0345
URL:http://www.microsoft.com/technet/security/bulletin/ms10-012.mspx

[ return to 127.0.0.23 ]

### 2.3.2 Medium 3389/tcp

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: SSL/TLS: Report Weak Cipher Suites
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: $Revision: 4863 $

**References**
CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000
Other:
URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html
URL:https://bettercrypto.org/

... continues on next page ...

... continued from previous page ...

URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: $Revision: 4781 $

**References**
Other:
URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with-sha-1-based-signature-algorithms/

[ return to 127.0.0.23 ]

### 2.3.3 Medium 135/tcp

| Medium (CVSS: 5.0)<br>NVT: DCE/RPC and MSRPC Services Enumeration Reporting |
|---|
| **Summary**<br>Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries. |
| **Vulnerability Detection Result**<br>Vulnerability was detected according to the Vulnerability Detection Method. |
| **Impact**<br>An attacker may use this fact to gain more knowledge about the remote host. |
| **Solution**<br>**Solution type:** Mitigation<br>Filter incoming traffic to this ports. |
| **Vulnerability Detection Method**<br>Details: `DCE/RPC and MSRPC Services Enumeration Reporting`<br>OID:1.3.6.1.4.1.25623.1.0.10736<br>Version used: `$Revision: 4998 $` |

[ return to 127.0.0.23 ]

### 2.3.4 Low general/tcp

| Low (CVSS: 2.6)<br>NVT: TCP timestamps |
|---|
| **Summary**<br>The remote host implements TCP timestamps and therefore allows to compute the uptime. |
| **Vulnerability Detection Result**<br>Vulnerability was detected according to the Vulnerability Detection Method. |
| **Impact**<br>A side effect of this feature is that the uptime of the remote host can sometimes be computed. |
| **Solution**<br>**Solution type:** Mitigation |

. . . continues on next page . . .

... continued from previous page ...

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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.23 ]

## 2.4 127.0.0.29

Host scan start Tue Feb 21 15:24:50 2017 UTC
Host scan end Tue Feb 21 15:35:42 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 3389/tcp | High |
| 445/tcp | High |

### 2.4.1 High 3389/tcp

High (CVSS: 9.3)
NVT: Microsoft Remote Desktop Protocol Remote Code Execution Vulnerabilities (2671387)

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS12-020.

... continues on next page ...

... continued from previous page ...

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation could allow remote attackers to execute arbitrary code as the logged-on user or cause a denial of service condition.

**Solution**
**Solution type:** VendorFix
Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory from the below link,
http://technet.microsoft.com/en-us/security/bulletin/ms12-020

**Affected Software/OS**
Microsoft Windows 7 Service Pack 1 and prior
Microsoft Windows XP Service Pack 3 and prior
Microsoft Windows 2K3 Service Pack 2 and prior
Microsoft Windows Vista Service Pack 2 and prior
Microsoft Windows Server 2008 Service Pack 2 and prior

**Vulnerability Insight**
The flaws are due to the way Remote Desktop Protocol accesses an object in memory that has been improperly initialized or has been deleted and the way RDP service processes the packets.

**Vulnerability Detection Method**
Details: `Microsoft Remote Desktop Protocol Remote Code Execution Vulnerabilities (267138.`
`↪..`
OID:1.3.6.1.4.1.25623.1.0.902818
Version used: `$Revision: 4234 $`

**References**
```
CVE: CVE-2012-0002, CVE-2012-0152
BID:52353, 52354
Other:
  URL:http://blog.binaryninjas.org/?p=58
   URL:http://secunia.com/advisories/48395
   URL:http://support.microsoft.com/kb/2671387
   URL:http://www.securitytracker.com/id/1026790
   URL:http://technet.microsoft.com/en-us/security/bulletin/ms12-020
```

[ return to 127.0.0.29 ]

### 2.4.2 High 445/tcp

High (CVSS: 0.0)
NVT: SMBv1 enabled (Remote Check)

**Summary**
The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.
This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices.

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

**Log Method**
Details: `SMBv1 enabled (Remote Check)`
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: `$Revision: 5222 $`

**References**
```
Other:
  URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best
↪-Practices
   URL:https://support.microsoft.com/en-us/kb/2696547
   URL:https://support.microsoft.com/en-us/kb/204279
   URL:https://technet.microsoft.com/en-us/library/security/MS17-010
```

High (CVSS: 10.0)
NVT: Vulnerabilities in SMB Could Allow Remote Code Execution (958687) - Remote

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS09-001.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation could allow remote unauthenticated attackers to cause denying the service by sending a specially crafted network message to a system running the server service.
Impact Level: System/Network

... continues on next page ...

... continued from previous page ...

**Solution**
**Solution type:** VendorFix
Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory from the below link, http://www.microsoft.com/technet/security/bulletin/ms09-001.mspx

**Affected Software/OS**
Microsoft Windows 2K Service Pack 4 and prior.
Microsoft Windows XP Service Pack 3 and prior.
Microsoft Windows 2003 Service Pack 2 and prior.

**Vulnerability Insight**
The issue is due to the way Server Message Block (SMB) Protocol software handles specially crafted SMB packets.

**Vulnerability Detection Method**
Details: `Vulnerabilities in SMB Could Allow Remote Code Execution (958687) - Remote`
OID:1.3.6.1.4.1.25623.1.0.900233
Version used: `$Revision: 4692 $`

**References**
`CVE: CVE-2008-4114, CVE-2008-4834, CVE-2008-4835`
`BID:31179`
`Other:`
`URL:http://www.milw0rm.com/exploits/6463`
`URL:http://www.microsoft.com/technet/security/bulletin/ms09-001.mspx`

High (CVSS: 10.0)
NVT: Microsoft Windows SMB Server NTLM Multiple Vulnerabilities (971468)

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS10-012.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow remote attackers to execute arbitrary code or cause a denial of service or bypass the authentication mechanism via brute force technique. Impact Level: System/Application

**Solution**
**Solution type:** VendorFix

... continues on next page ...

... continued from previous page ...

Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory from the below link, http://www.microsoft.com/technet/security/bulletin/ms10-012.mspx

**Affected Software/OS**
Microsoft Windows 7 Microsoft Windows 2000 Service Pack and prior Microsoft Windows XP Service Pack 3 and prior Microsoft Windows Vista Service Pack 2 and prior Microsoft Windows Server 2003 Service Pack 2 and prior Microsoft Windows Server 2008 Service Pack 2 and prior

**Vulnerability Insight**
- An input validation error exists while processing SMB requests and can be exploited to cause a buffer overflow via a specially crafted SMB packet. - An error exists in the SMB implementation while parsing SMB packets during the Negotiate phase causing memory corruption via a specially crafted SMB packet. - NULL pointer dereference error exists in SMB while verifying the 'share' and 'servername' fields in SMB packets causing denial of service. - A lack of cryptographic entropy when the SMB server generates challenges during SMB NTLM authentication and can be exploited to bypass the authentication mechanism.

**Vulnerability Detection Method**
Details: `Microsoft Windows SMB Server NTLM Multiple Vulnerabilities (971468)`
OID:1.3.6.1.4.1.25623.1.0.902269
Version used: `$Revision: 5136 $`

**References**
```
CVE: CVE-2010-0020, CVE-2010-0021, CVE-2010-0022, CVE-2010-0231
Other:
  URL:http://secunia.com/advisories/38510/
   URL:http://support.microsoft.com/kb/971468
   URL:http://www.vupen.com/english/advisories/2010/0345
   URL:http://www.microsoft.com/technet/security/bulletin/ms10-012.mspx
```

[ return to 127.0.0.29 ]

## 2.5 127.0.0.14

Host scan start Tue Feb 21 15:24:51 2017 UTC
Host scan end Tue Feb 21 15:57:20 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 445/tcp | High |
| 389/tcp | Medium |
| 80/tcp | Medium |
| 3389/tcp | Medium |
| 3269/tcp | Medium |
| 3268/tcp | Medium |
| 135/tcp | Medium |

... (continues) ...

... (continued) ...

| Service (Port) | Threat Level |
|---|---|
| 443/tcp | Medium |
| 636/tcp | Medium |
| general/tcp | Low |

### 2.5.1 High 445/tcp

High (CVSS: 0.0)
NVT: SMBv1 enabled (Remote Check)

**Summary**
The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.
This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices.

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

**Log Method**
Details: `SMBv1 enabled (Remote Check)`
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: `$Revision: 5222 $`

**References**
```
Other:
  URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best
↪-Practices
   URL:https://support.microsoft.com/en-us/kb/2696547
   URL:https://support.microsoft.com/en-us/kb/204279
   URL:https://technet.microsoft.com/en-us/library/security/MS17-010
```

[ return to 127.0.0.14 ]

### 2.5.2 Medium 389/tcp

Medium (CVSS: 5.0)
NVT: Use LDAP search request to retrieve information from NT Directory Services

**Summary**
It is possible to disclose LDAP information.
Description :
The directory base of the remote server is set to NULL. This allows information to be enumerated without any prior knowledge of the directory structure.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
If pre-Windows 2000 compatibility is not required, remove pre-Windows 2000 compatibility as follows :
- start cmd.exe
- execute the command : net localgroup 'Pre-Windows 2000 Compatible Access' everyone /delete
- restart the remote host

**Vulnerability Detection Method**
Details: `Use LDAP search request to retrieve information from NT Directory Services`
OID:1.3.6.1.4.1.25623.1.0.12105
Version used: `$Revision: 5190 $`

[ return to 127.0.0.14 ]

### 2.5.3 Medium 80/tcp

Medium (CVSS: 5.0)
NVT: Microsoft IIS Default Welcome Page Information Disclosure Vulnerability

**Summary**
The host is running Microsoft IIS Webserver and is prone to information disclosure vulnerability.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow remote attackers to obtain sensitive information that could aid in further attacks.

**Solution**
**Solution type:** Mitigation
Disable the default pages within the server configuration.

**Affected Software/OS**

. . . continues on next page . . .

... continued from previous page ...

Microsoft Internet Information Services

**Vulnerability Insight**
The flaw is due to misconfiguration of IIS Server, which allows to access default pages when the server is not used.

**Vulnerability Detection Method**
Details: `Microsoft IIS Default Welcome Page Information Disclosure Vulnerability`
OID:1.3.6.1.4.1.25623.1.0.802806
Version used: `$Revision: 2715 $`

**References**
```
Other:
  URL:http://www.iis.net/
```

[ return to 127.0.0.14 ]

### 2.5.4 Medium 3389/tcp

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

... continues on next page ...

... continued from previous page ...

**Vulnerability Detection Method**
Details: SSL/TLS: Report Weak Cipher Suites
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: $Revision: 4863 $

**References**
CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000
Other:
URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html
URL:https://bettercrypto.org/
URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/

Medium (CVSS: 4.0)
NVT: SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerability

**Summary**
The SSL/TLS service uses Diffie-Hellman groups with insufficient strength (key size < 2048).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to decrypt the SSL/TLS communication offline.

**Solution**
**Solution type:** Workaround
Deploy (Ephemeral) Elliptic-Curve Diffie-Hellman (ECDHE) or use a 2048-bit or stronger Diffie-Hellman group. (see https://weakdh.org/sysadmin.html).
For Apache Web Servers: Beginning with version 2.4.7, mod_ssl will use DH parameters which include primes with lengths of more than 1024 bits.

**Vulnerability Insight**
The Diffie-Hellman group are some big numbers that are used as base for the DH computations. They can be, and often are, fixed. The security of the final secret depends on the size of these parameters. It was found that 512 and 768 bits to be weak, 1024 bits to be breakable by really powerful attackers like governments.

**Vulnerability Detection Method**
Checks the DHE temporary public key size.
Details: SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerabili..
OID:1.3.6.1.4.1.25623.1.0.106223
Version used: $Revision: 4739 $

... continues on next page ...

... continued from previous page ...

**References**
Other:
URL:https://weakdh.org/
URL:https://weakdh.org/sysadmin.html

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: $Revision: 4781 $

**References**
Other:
URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with

... continues on next page ...

... continued from previous page ...

↪-sha-1-based-signature-algorithms/

[ return to 127.0.0.14 ]

### 2.5.5 Medium 3269/tcp

Medium (CVSS: 5.0)
NVT: Use LDAP search request to retrieve information from NT Directory Services

**Summary**
It is possible to disclose LDAP information.
Description :
The directory base of the remote server is set to NULL. This allows information to be enumerated without any prior knowledge of the directory structure.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
If pre-Windows 2000 compatibility is not required, remove pre-Windows 2000 compatibility as follows :
- start cmd.exe
- execute the command : net localgroup 'Pre-Windows 2000 Compatible Access' everyone /delete
- restart the remote host

**Vulnerability Detection Method**
Details: `Use LDAP search request to retrieve information from NT Directory Services`
OID:1.3.6.1.4.1.25623.1.0.12105
Version used: `$Revision: 5190 $`

Medium (CVSS: 4.3)
NVT: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection

**Summary**
It was possible to detect the usage of the deprecated SSLv2 and/or SSLv3 protocol on this system.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to use the known cryptographic flaws to eavesdrop the connection between clients and the service to get access to sensitive data transferred within the secured connection.

... continues on next page ...

... continued from previous page ...

**Solution**
**Solution type:** Mitigation
It is recommended to disable the deprecated SSLv2 and/or SSLv3 protocols in favor of the TLSv1+ protocols. Please see the references for more information.

**Affected Software/OS**
All services providing an encrypted communication using the SSLv2 and/or SSLv3 protocols.

**Vulnerability Insight**
The SSLv2 and SSLv3 protocols containing known cryptographic flaws like:
- Padding Oracle On Downgraded Legacy Encryption (POODLE, CVE-2014-3566)
- Decrypting RSA with Obsolete and Weakened eNcryption (DROWN, CVE-2016-0800)

**Vulnerability Detection Method**
Check the used protocols of the services provided by this system.
Details: `SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection`
OID:1.3.6.1.4.1.25623.1.0.111012
Version used: `$Revision: 4686 $`

**References**
```
CVE: CVE-2016-0800, CVE-2014-3566
Other:
  URL:https://www.enisa.europa.eu/activities/identity-and-trust/library/delivera
↪bles/algorithms-key-sizes-and-parameters-report
   URL:https://bettercrypto.org/
   URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
   URL:https://drownattack.com/
   URL:https://www.imperialviolet.org/2014/10/14/poodle.html
```

Medium (CVSS: 4.3)
NVT: SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability (POODLE)

**Summary**
This host is prone to an information disclosure vulnerability.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow a man-in-the-middle attackers gain access to the plain text data stream.

**Solution**
**Solution type:** Mitigation

... continues on next page ...

... continued from previous page ...

Possible Mitigations are:
- Disable SSLv3
- Disable cipher suites supporting CBC cipher modes
- Enable TLS_FALLBACK_SCSV if the service is providing TLSv1.0+

**Vulnerability Insight**
The flaw is due to the block cipher padding not being deterministic and not covered by the Message Authentication Code

**Vulnerability Detection Method**
Evaluate previous collected information about this service.
Details: `SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability ..`
OID:1.3.6.1.4.1.25623.1.0.802087
Version used: `$Revision: 4749 $`

**References**
`CVE: CVE-2014-3566`
`BID:70574`
`Other:`
`URL:https://www.openssl.org/~bodo/ssl-poodle.pdf`
`URL:https://www.imperialviolet.org/2014/10/14/poodle.html`
`URL:https://www.dfranke.us/posts/2014-10-14-how-poodle-happened.html`
`URL:http://googleonlinesecurity.blogspot.in/2014/10/this-poodle-bites-exploiting-ssl-30.html`

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).

... continues on next page ...

... continued from previous page ...

- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Weak Cipher Suites`
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: `$Revision: 4863 $`

**References**
`CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000`
`Other:`
`URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html`
`URL:https://bettercrypto.org/`
`URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/`

Medium (CVSS: 4.0)
NVT: SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerability

**Summary**
The SSL/TLS service uses Diffie-Hellman groups with insufficient strength (key size < 2048).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to decrypt the SSL/TLS communication offline.

**Solution**
**Solution type:** Workaround
Deploy (Ephemeral) Elliptic-Curve Diffie-Hellman (ECDHE) or use a 2048-bit or stronger Diffie-Hellman group. (see https://weakdh.org/sysadmin.html).
For Apache Web Servers: Beginning with version 2.4.7, mod_ssl will use DH parameters which include primes with lengths of more than 1024 bits.

**Vulnerability Insight**
The Diffie-Hellman group are some big numbers that are used as base for the DH computations. They can be, and often are, fixed. The security of the final secret depends on the size of these parameters. It was found that 512 and 768 bits to be weak, 1024 bits to be breakable by really powerful attackers like governments.

**Vulnerability Detection Method**
Checks the DHE temporary public key size.

... continues on next page ...

... continued from previous page ...

Details: SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerabili.
↪..
OID:1.3.6.1.4.1.25623.1.0.106223
Version used: $Revision: 4739 $

**References**
Other:
URL:https://weakdh.org/
URL:https://weakdh.org/sysadmin.html

[ return to 127.0.0.14 ]

### 2.5.6 Medium 3268/tcp

Medium (CVSS: 5.0)
NVT: Use LDAP search request to retrieve information from NT Directory Services

**Summary**
It is possible to disclose LDAP information.
Description :
The directory base of the remote server is set to NULL. This allows information to be enumerated without any prior knowledge of the directory structure.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
If pre-Windows 2000 compatibility is not required, remove pre-Windows 2000 compatibility as follows :
- start cmd.exe
- execute the command : net localgroup 'Pre-Windows 2000 Compatible Access' everyone /delete
- restart the remote host

**Vulnerability Detection Method**
Details: Use LDAP search request to retrieve information from NT Directory Services
OID:1.3.6.1.4.1.25623.1.0.12105
Version used: $Revision: 5190 $

[ return to 127.0.0.14 ]

### 2.5.7 Medium 135/tcp

Medium (CVSS: 5.0)
NVT: DCE/RPC and MSRPC Services Enumeration Reporting

**Summary**
Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker may use this fact to gain more knowledge about the remote host.

**Solution**
**Solution type:** Mitigation
Filter incoming traffic to this ports.

**Vulnerability Detection Method**
Details: `DCE/RPC and MSRPC Services Enumeration Reporting`
OID:1.3.6.1.4.1.25623.1.0.10736
Version used: `$Revision: 4998 $`

[ return to 127.0.0.14 ]

### 2.5.8 Medium 443/tcp

Medium (CVSS: 5.0)
NVT: Microsoft IIS Default Welcome Page Information Disclosure Vulnerability

**Summary**
The host is running Microsoft IIS Webserver and is prone to information disclosure vulnerability.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow remote attackers to obtain sensitive information that could aid in further attacks.

**Solution**
**Solution type:** Mitigation
Disable the default pages within the server configuration.

**Affected Software/OS**
Microsoft Internet Information Services

. . . continues on next page . . .

... continued from previous page ...

**Vulnerability Insight**
The flaw is due to misconfiguration of IIS Server, which allows to access default pages when the server is not used.

**Vulnerability Detection Method**
Details: `Microsoft IIS Default Welcome Page Information Disclosure Vulnerability`
OID:1.3.6.1.4.1.25623.1.0.802806
Version used: `$Revision: 2715 $`

**References**
`Other:`
`URL:http://www.iis.net/`

Medium (CVSS: 5.0)
NVT: SSL/TLS: Report Vulnerable Cipher Suites for HTTPS

**Summary**
This routine reports all SSL/TLS cipher suites accepted by a service where attack vectors exists only on HTTPS services.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Affected Software/OS**
Services accepting vulnerable SSL/TLS cipher suites via HTTPS.

**Vulnerability Insight**
These rules are applied for the evaluation of the vulnerable cipher suites:
- 64-bit block cipher 3DES vulnerable to the SWEET32 attack (CVE-2016-2183).

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Vulnerable Cipher Suites for HTTPS`
OID:1.3.6.1.4.1.25623.1.0.108031
Version used: `$Revision: 5232 $`

**References**
`CVE: CVE-2016-2183, CVE-2016-6329`
`Other:`
`URL:https://bettercrypto.org/`

... continues on next page ...

... continued from previous page ...

URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
URL:https://sweet32.info/

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: SSL/TLS: Report Weak Cipher Suites
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: $Revision: 4863 $

**References**
CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000
Other:
URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html
URL:https://bettercrypto.org/
URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/

Medium (CVSS: 4.3)
NVT: SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability (POODLE)

**Summary**
This host is prone to an information disclosure vulnerability.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow a man-in-the-middle attackers gain access to the plain text data stream.

**Solution**
**Solution type:** Mitigation
Possible Mitigations are:
- Disable SSLv3
- Disable cipher suites supporting CBC cipher modes
- Enable TLS_FALLBACK_SCSV if the service is providing TLSv1.0+

**Vulnerability Insight**
The flaw is due to the block cipher padding not being deterministic and not covered by the Message Authentication Code

**Vulnerability Detection Method**
Evaluate previous collected information about this service.
Details: `SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability ..`
OID:1.3.6.1.4.1.25623.1.0.802087
Version used: `$Revision: 4749 $`

**References**
```
CVE: CVE-2014-3566
BID:70574
Other:
  URL:https://www.openssl.org/~bodo/ssl-poodle.pdf
   URL:https://www.imperialviolet.org/2014/10/14/poodle.html
   URL:https://www.dfranke.us/posts/2014-10-14-how-poodle-happened.html
   URL:http://googleonlinesecurity.blogspot.in/2014/10/this-poodle-bites-exploiting-ssl-30.html
```

Medium (CVSS: 4.3)
NVT: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection

**Summary**

. . . continues on next page . . .

... continued from previous page ...

It was possible to detect the usage of the deprecated SSLv2 and/or SSLv3 protocol on this system.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to use the known cryptographic flaws to eavesdrop the connection between clients and the service to get access to sensitive data transferred within the secured connection.

**Solution**
**Solution type:** Mitigation
It is recommended to disable the deprecated SSLv2 and/or SSLv3 protocols in favor of the TLSv1+ protocols. Please see the references for more information.

**Affected Software/OS**
All services providing an encrypted communication using the SSLv2 and/or SSLv3 protocols.

**Vulnerability Insight**
The SSLv2 and SSLv3 protocols containing known cryptographic flaws like:
- Padding Oracle On Downgraded Legacy Encryption (POODLE, CVE-2014-3566)
- Decrypting RSA with Obsolete and Weakened eNcryption (DROWN, CVE-2016-0800)

**Vulnerability Detection Method**
Check the used protocols of the services provided by this system.
Details: `SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection`
OID:1.3.6.1.4.1.25623.1.0.111012
Version used: `$Revision: 4686 $`

**References**
```
CVE: CVE-2016-0800, CVE-2014-3566
Other:
  URL:https://www.enisa.europa.eu/activities/identity-and-trust/library/delivera
↪bles/algorithms-key-sizes-and-parameters-report
   URL:https://bettercrypto.org/
   URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
   URL:https://drownattack.com/
   URL:https://www.imperialviolet.org/2014/10/14/poodle.html
```

Medium (CVSS: 4.0)
NVT: SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerability

**Summary**
The SSL/TLS service uses Diffie-Hellman groups with insufficient strength (key size < 2048).

... continues on next page ...

... continued from previous page ...

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to decrypt the SSL/TLS communication offline.

**Solution**
**Solution type:** Workaround
Deploy (Ephemeral) Elliptic-Curve Diffie-Hellman (ECDHE) or use a 2048-bit or stronger Diffie-Hellman group. (see https://weakdh.org/sysadmin.html).
For Apache Web Servers: Beginning with version 2.4.7, mod_ssl will use DH parameters which include primes with lengths of more than 1024 bits.

**Vulnerability Insight**
The Diffie-Hellman group are some big numbers that are used as base for the DH computations. They can be, and often are, fixed. The security of the final secret depends on the size of these parameters. It was found that 512 and 768 bits to be weak, 1024 bits to be breakable by really powerful attackers like governments.

**Vulnerability Detection Method**
Checks the DHE temporary public key size.
Details: `SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerabili.`
`↪..`
OID:1.3.6.1.4.1.25623.1.0.106223
Version used: `$Revision: 4739 $`

**References**
```
Other:
  URL:https://weakdh.org/
   URL:https://weakdh.org/sysadmin.html
```

[ return to 127.0.0.14 ]

### 2.5.9 Medium 636/tcp

Medium (CVSS: 5.0)
NVT: Use LDAP search request to retrieve information from NT Directory Services

**Summary**
It is possible to disclose LDAP information.
Description :
The directory base of the remote server is set to NULL. This allows information to be enumerated without any prior knowledge of the directory structure.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

... continues on next page ...

... continued from previous page ...

**Solution**
**Solution type:** Workaround
If pre-Windows 2000 compatibility is not required, remove pre-Windows 2000 compatibility as follows :
- start cmd.exe
- execute the command : net localgroup 'Pre-Windows 2000 Compatible Access' everyone /delete
- restart the remote host

**Vulnerability Detection Method**
Details: `Use LDAP search request to retrieve information from NT Directory Services`
OID:1.3.6.1.4.1.25623.1.0.12105
Version used: `$Revision: 5190 $`

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Weak Cipher Suites`
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: `$Revision: 4863 $`

**References**

... continues on next page ...

... continued from previous page ...

CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000
Other:
URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html
URL:https://bettercrypto.org/
URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/

Medium (CVSS: 4.3)
NVT: SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability (POODLE)

**Summary**
This host is prone to an information disclosure vulnerability.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow a man-in-the-middle attackers gain access to the plain text data stream.

**Solution**
**Solution type:** Mitigation
Possible Mitigations are:
- Disable SSLv3
- Disable cipher suites supporting CBC cipher modes
- Enable TLS_FALLBACK_SCSV if the service is providing TLSv1.0+

**Vulnerability Insight**
The flaw is due to the block cipher padding not being deterministic and not covered by the Message Authentication Code

**Vulnerability Detection Method**
Evaluate previous collected information about this service.
Details: SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability ...
OID:1.3.6.1.4.1.25623.1.0.802087
Version used: $Revision: 4749 $

**References**
CVE: CVE-2014-3566
BID:70574
Other:
URL:https://www.openssl.org/~bodo/ssl-poodle.pdf
URL:https://www.imperialviolet.org/2014/10/14/poodle.html
URL:https://www.dfranke.us/posts/2014-10-14-how-poodle-happened.html
URL:http://googleonlinesecurity.blogspot.in/2014/10/this-poodle-bites-exploit

... continues on next page ...

... continued from previous page ...

↪ing-ssl-30.html

Medium (CVSS: 4.3)
NVT: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection

**Summary**
It was possible to detect the usage of the deprecated SSLv2 and/or SSLv3 protocol on this system.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to use the known cryptographic flaws to eavesdrop the connection between clients and the service to get access to sensitive data transferred within the secured connection.

**Solution**
**Solution type:** Mitigation
It is recommended to disable the deprecated SSLv2 and/or SSLv3 protocols in favor of the TLSv1+ protocols. Please see the references for more information.

**Affected Software/OS**
All services providing an encrypted communication using the SSLv2 and/or SSLv3 protocols.

**Vulnerability Insight**
The SSLv2 and SSLv3 protocols containing known cryptographic flaws like:
- Padding Oracle On Downgraded Legacy Encryption (POODLE, CVE-2014-3566)
- Decrypting RSA with Obsolete and Weakened eNcryption (DROWN, CVE-2016-0800)

**Vulnerability Detection Method**
Check the used protocols of the services provided by this system.
Details: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection
OID:1.3.6.1.4.1.25623.1.0.111012
Version used: $Revision: 4686 $

**References**
CVE: CVE-2016-0800, CVE-2014-3566
Other:
URL:https://www.enisa.europa.eu/activities/identity-and-trust/library/deliverables/algorithms-key-sizes-and-parameters-report
URL:https://bettercrypto.org/
URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
URL:https://drownattack.com/
URL:https://www.imperialviolet.org/2014/10/14/poodle.html

Medium (CVSS: 4.0)
NVT: SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerability

**Summary**
The SSL/TLS service uses Diffie-Hellman groups with insufficient strength (key size < 2048).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to decrypt the SSL/TLS communication offline.

**Solution**
**Solution type:** Workaround
Deploy (Ephemeral) Elliptic-Curve Diffie-Hellman (ECDHE) or use a 2048-bit or stronger Diffie-Hellman group. (see https://weakdh.org/sysadmin.html).
For Apache Web Servers: Beginning with version 2.4.7, mod_ssl will use DH parameters which include primes with lengths of more than 1024 bits.

**Vulnerability Insight**
The Diffie-Hellman group are some big numbers that are used as base for the DH computations. They can be, and often are, fixed. The security of the final secret depends on the size of these parameters. It was found that 512 and 768 bits to be weak, 1024 bits to be breakable by really powerful attackers like governments.

**Vulnerability Detection Method**
Checks the DHE temporary public key size.
Details: `SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerabili.`
`↪..`
OID:1.3.6.1.4.1.25623.1.0.106223
Version used: `$Revision: 4739 $`

**References**
```
Other:
  URL:https://weakdh.org/
   URL:https://weakdh.org/sysadmin.html
```

[ return to 127.0.0.14 ]

### 2.5.10 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

. . . continues on next page . . .

... continued from previous page ...

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.14 ]

## 2.6 127.0.0.1

Host scan start Tue Feb 21 15:24:50 2017 UTC
Host scan end Tue Feb 21 16:29:33 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 445/tcp | High |
| 22/tcp | High |
| 135/tcp | Medium |

... (continues) ...

... (continued) ...

| Service (Port) | Threat Level |
|---|---|
| 22/tcp | Medium |
| 3389/tcp | Medium |
| 22/tcp | Low |
| general/tcp | Low |

### 2.6.1 High 445/tcp

High (CVSS: 0.0)
NVT: SMBv1 enabled (Remote Check)

**Summary**
The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.
This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices.

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

**Log Method**
Details: `SMBv1 enabled (Remote Check)`
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: `$Revision: 5222 $`

**References**
```
Other:
  URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best
↪-Practices
   URL:https://support.microsoft.com/en-us/kb/2696547
   URL:https://support.microsoft.com/en-us/kb/204279
   URL:https://technet.microsoft.com/en-us/library/security/MS17-010
```

High (CVSS: 10.0)
NVT: Microsoft Windows SMB Server NTLM Multiple Vulnerabilities (971468)

... continues on next page ...

... continued from previous page ...

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS10-012.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow remote attackers to execute arbitrary code or cause a denial of service or bypass the authentication mechanism via brute force technique. Impact Level: System/Application

**Solution**
**Solution type:** VendorFix
Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory from the below link, http://www.microsoft.com/technet/security/bulletin/ms10-012.mspx

**Affected Software/OS**
Microsoft Windows 7 Microsoft Windows 2000 Service Pack and prior Microsoft Windows XP Service Pack 3 and prior Microsoft Windows Vista Service Pack 2 and prior Microsoft Windows Server 2003 Service Pack 2 and prior Microsoft Windows Server 2008 Service Pack 2 and prior

**Vulnerability Insight**
- An input validation error exists while processing SMB requests and can be exploited to cause a buffer overflow via a specially crafted SMB packet. - An error exists in the SMB implementation while parsing SMB packets during the Negotiate phase causing memory corruption via a specially crafted SMB packet. - NULL pointer dereference error exists in SMB while verifying the 'share' and 'servername' fields in SMB packets causing denial of service. - A lack of cryptographic entropy when the SMB server generates challenges during SMB NTLM authentication and can be exploited to bypass the authentication mechanism.

**Vulnerability Detection Method**
Details: `Microsoft Windows SMB Server NTLM Multiple Vulnerabilities (971468)`
OID:1.3.6.1.4.1.25623.1.0.902269
Version used: `$Revision: 5136 $`

**References**
`CVE: CVE-2010-0020, CVE-2010-0021, CVE-2010-0022, CVE-2010-0231`
`Other:`
`URL:http://secunia.com/advisories/38510/`
`URL:http://support.microsoft.com/kb/971468`
`URL:http://www.vupen.com/english/advisories/2010/0345`
`URL:http://www.microsoft.com/technet/security/bulletin/ms10-012.mspx`

[ return to 127.0.0.1 ]

### 2.6.2 High 22/tcp

High (CVSS: 7.8)
NVT: OpenSSH Denial of Service And User Enumeration Vulnerabilities (Windows)

**Summary**
This host is installed with openssh and is prone to denial of service and user enumeration vulnerabilities.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successfully exploiting this issue allows remote attackers to cause a denial of service (crypt CPU consumption) and to enumerate users by leveraging the timing difference between responses when a large password is provided.

**Solution**
**Solution type:** VendorFix
Upgrade to OpenSSH version 7.3 or later. For updates refer to http://www.openssh.com

**Affected Software/OS**
OpenSSH versions before 7.3 on Windows

**Vulnerability Insight**
Multiple flaws exists due to,
- The auth_password function in 'auth-passwd.c' script does not limit password lengths for password authentication.
- The sshd in OpenSSH, when SHA256 or SHA512 are used for user password hashing uses BLOWFISH hashing on a static password when the username does not exist and it takes much longer to calculate SHA256/SHA512 hash than BLOWFISH hash.

**Vulnerability Detection Method**
Checks if a vulnerable version is present on the target host.
Details: `OpenSSH Denial of Service And User Enumeration Vulnerabilities (Windows)`
OID:1.3.6.1.4.1.25623.1.0.809121
Version used: `$Revision: 5083 $`

**References**
`CVE: CVE-2016-6515, CVE-2016-6210`
`BID:92212`
`Other:`
`URL:http://www.openssh.com/txt/release-7.3`
`URL:http://seclists.org/fulldisclosure/2016/Jul/51`
`URL:https://security-tracker.debian.org/tracker/CVE-2016-6210`
`URL:http://openwall.com/lists/oss-security/2016/08/01/2`

... continues on next page ...

... continued from previous page ...

High (CVSS: 7.5)
NVT: OpenSSH Multiple Vulnerabilities Jan17 (Windows)

**Summary**
This host is installed with openssh and is prone to multiple vulnerabilities.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successfully exploiting this issue allows local users to obtain sensitive private-key information, to gain privileges, conduct a senial-of-service condition and allows remote attackers to execute arbitrary local PKCS#11 modules.
Impact Level: Application

**Solution**
**Solution type:** VendorFix
Upgrade to OpenSSH version 7.4 or later. For updates refer to http://www.openssh.com

**Affected Software/OS**
OpenSSH versions before 7.4 on Windows

**Vulnerability Insight**
Multiple flaws exists due to, - An 'authfile.c' script does not properly consider the effects of realloc on buffer contents. - The shared memory manager (associated with pre-authentication compression) does not ensure that a bounds check is enforced by all compilers. - The sshd in OpenSSH creates forwarded Unix-domain sockets as root, when privilege separation is not used. - An untrusted search path vulnerability in ssh-agent.c in ssh-agent. - NULL pointer dereference error due to an out-of-sequence NEWKEYS message.

**Vulnerability Detection Method**
Get the installed version with the help of detect NVT and check the version is vulnerable or not.
Details: `OpenSSH Multiple Vulnerabilities Jan17 (Windows)`
OID:1.3.6.1.4.1.25623.1.0.810325
Version used: `$Revision: 5084 $`

**References**
```
CVE: CVE-2016-10009, CVE-2016-10010, CVE-2016-10011, CVE-2016-10012, CVE-2016-10
↪708
BID:94968, 94972, 94977, 94975
Other:
  URL:https://www.openssh.com/txt/release-7.4
   URL:http://www.openwall.com/lists/oss-security/2016/12/19/2
   URL:http://blog.swiecki.net/2018/01/fuzzing-tcp-servers.html
   URL:https://anongit.mindrot.org/openssh.git/commit/?id=28652bca29046f62c7045e
↪933e6b931de1d16737
```

[ return to 127.0.0.1 ]

### 2.6.3 Medium 135/tcp

| Medium (CVSS: 5.0)<br>NVT: DCE/RPC and MSRPC Services Enumeration Reporting |
|---|
| **Summary**<br>Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries. |
| **Vulnerability Detection Result**<br>Vulnerability was detected according to the Vulnerability Detection Method. |
| **Impact**<br>An attacker may use this fact to gain more knowledge about the remote host. |
| **Solution**<br>**Solution type:** Mitigation<br>Filter incoming traffic to this ports. |
| **Vulnerability Detection Method**<br>Details: `DCE/RPC and MSRPC Services Enumeration Reporting`<br>OID:1.3.6.1.4.1.25623.1.0.10736<br>Version used: `$Revision: 4998 $` |

[ return to 127.0.0.1 ]

### 2.6.4 Medium 22/tcp

| Medium (CVSS: 4.3)<br>NVT: SSH Weak Encryption Algorithms Supported |
|---|
| **Summary**<br>The remote SSH server is configured to allow weak encryption algorithms. |
| **Vulnerability Detection Result**<br>Vulnerability was detected according to the Vulnerability Detection Method. |
| **Solution**<br>**Solution type:** Mitigation<br>Disable the weak encryption algorithms. |
| **Vulnerability Insight** |
| . . . continues on next page . . . |

... continued from previous page ...

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: 4490 $`

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

[ return to 127.0.0.1 ]

### 2.6.5 Medium 3389/tcp

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).

... continues on next page ...

... continued from previous page ...

- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Weak Cipher Suites`
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: `$Revision: 4863 $`

**References**
`CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000`
`Other:`
`URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html`
`URL:https://bettercrypto.org/`
`URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/`

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1

... continues on next page ...

... continued from previous page ...

or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: `SSL/TLS: Certificate Signed Using A Weak Signature Algorithm`
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: `$Revision: 4781 $`

**References**
```
Other:
  URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with
↪-sha-1-based-signature-algorithms/
```

[ return to 127.0.0.1 ]

### 2.6.6 Low 22/tcp

Low (CVSS: 2.6)
NVT: SSH Weak MAC Algorithms Supported

**Summary**
The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Disable the weak MAC algorithms.

**Vulnerability Detection Method**
Details: `SSH Weak MAC Algorithms Supported`
OID:1.3.6.1.4.1.25623.1.0.105610
Version used: `$Revision: 4490 $`

[ return to 127.0.0.1 ]

### 2.6.7 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

**Summary**

... continues on next page ...

... continued from previous page ...

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $`

**References**
```
Other:
  URL:http://www.ietf.org/rfc/rfc1323.txt
```

[ return to 127.0.0.1 ]

## 2.7 127.0.0.10

Host scan start Tue Feb 21 15:24:50 2017 UTC
Host scan end Tue Feb 21 16:29:11 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 445/tcp | High |
| 22/tcp | High |

... (continues) ...

... (continued) ...

| Service (Port) | Threat Level |
|---|---|
| 3389/tcp | Medium |
| 22/tcp | Medium |
| 135/tcp | Medium |
| general/tcp | Low |
| 22/tcp | Low |

### 2.7.1 High 445/tcp

High (CVSS: 10.0)
NVT: Microsoft Windows SMB2 Negotiation Protocol Remote Code Execution Vulnerability

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS09-050.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker can exploit this issue to execute code with SYSTEM-level privileges failed exploit attempts will likely cause denial-of-service conditions.
Impact Level: System

**Solution**
**Solution type:** VendorFix

**Affected Software/OS**
- Windows 7 RC
- Windows Vista and
- Windows 2008 Server

**Vulnerability Insight**
Multiple vulnerabilities exists,
- A denial of service vulnerability exists in the way that Microsoft Server Message Block (SMB) Protocol software handles specially crafted SMB version 2 (SMBv2) packets.
- Unauthenticated remote code execution vulnerability exists in the way that Microsoft Server Message Block (SMB) Protocol software handles specially crafted SMB packets.

**Vulnerability Detection Method**
Details: `Microsoft Windows SMB2 Negotiation Protocol Remote Code Execution Vulnerability`
OID:1.3.6.1.4.1.25623.1.0.900965
Version used: `$Revision: 5074 $`

**References**
`CVE: CVE-2009-2526, CVE-2009-2532, CVE-2009-3103`
`BID:36299`

... continues on next page ...

... continued from previous page ...

```
Other:
  URL:http://www.microsoft.com/technet/security/bulletin/MS09-050.mspx
```

**Note**

```
This is a sample note on this scan result which I would like to see for any othe
↪r occurance of
this vulnerability, regardless of the task or host.
```

Last modified: Thu Mar 23 16:52:39 2017 UTC

High (CVSS: 10.0)
NVT: Microsoft Windows SMB Server NTLM Multiple Vulnerabilities (971468)

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS10-012.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow remote attackers to execute arbitrary code or cause a denial of service or bypass the authentication mechanism via brute force technique. Impact Level: System/Application

**Solution**
**Solution type:** VendorFix
Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory from the below link, http://www.microsoft.com/technet/security/bulletin/ms10-012.mspx

**Affected Software/OS**
Microsoft Windows 7 Microsoft Windows 2000 Service Pack and prior Microsoft Windows XP Service Pack 3 and prior Microsoft Windows Vista Service Pack 2 and prior Microsoft Windows Server 2003 Service Pack 2 and prior Microsoft Windows Server 2008 Service Pack 2 and prior

**Vulnerability Insight**
- An input validation error exists while processing SMB requests and can be exploited to cause a buffer overflow via a specially crafted SMB packet. - An error exists in the SMB implementation while parsing SMB packets during the Negotiate phase causing memory corruption via a specially crafted SMB packet. - NULL pointer dereference error exists in SMB while verifying the 'share' and 'servername' fields in SMB packets causing denial of service. - A lack of cryptographic entropy when the SMB server generates challenges during SMB NTLM authentication and can be exploited to bypass the authentication mechanism.

... continues on next page ...

... continued from previous page ...

**Vulnerability Detection Method**
Details: Microsoft Windows SMB Server NTLM Multiple Vulnerabilities (971468)
OID:1.3.6.1.4.1.25623.1.0.902269
Version used: $Revision: 5136 $

**References**
CVE: CVE-2010-0020, CVE-2010-0021, CVE-2010-0022, CVE-2010-0231
Other:
URL:http://secunia.com/advisories/38510/
URL:http://support.microsoft.com/kb/971468
URL:http://www.vupen.com/english/advisories/2010/0345
URL:http://www.microsoft.com/technet/security/bulletin/ms10-012.mspx

High (CVSS: 0.0)
NVT: SMBv1 enabled (Remote Check)

**Summary**
The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.
This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices.

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

**Log Method**
Details: SMBv1 enabled (Remote Check)
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: $Revision: 5222 $

**References**
Other:
URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best-Practices
URL:https://support.microsoft.com/en-us/kb/2696547
URL:https://support.microsoft.com/en-us/kb/204279
URL:https://technet.microsoft.com/en-us/library/security/MS17-010

[ return to 127.0.0.10 ]

### 2.7.2 High 22/tcp

High (CVSS: 7.8)
NVT: OpenSSH Denial of Service And User Enumeration Vulnerabilities (Windows)

**Summary**
This host is installed with openssh and is prone to denial of service and user enumeration vulnerabilities.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successfully exploiting this issue allows remote attackers to cause a denial of service (crypt CPU consumption) and to enumerate users by leveraging the timing difference between responses when a large password is provided.

**Solution**
**Solution type:** VendorFix
Upgrade to OpenSSH version 7.3 or later. For updates refer to http://www.openssh.com

**Affected Software/OS**
OpenSSH versions before 7.3 on Windows

**Vulnerability Insight**
Multiple flaws exists due to,
- The auth_password function in 'auth-passwd.c' script does not limit password lengths for password authentication.
- The sshd in OpenSSH, when SHA256 or SHA512 are used for user password hashing uses BLOWFISH hashing on a static password when the username does not exist and it takes much longer to calculate SHA256/SHA512 hash than BLOWFISH hash.

**Vulnerability Detection Method**
Checks if a vulnerable version is present on the target host.
Details: `OpenSSH Denial of Service And User Enumeration Vulnerabilities (Windows)`
OID:1.3.6.1.4.1.25623.1.0.809121
Version used: `$Revision: 5083 $`

**References**
`CVE: CVE-2016-6515, CVE-2016-6210`
`BID:92212`
`Other:`
`URL:http://www.openssh.com/txt/release-7.3`
`URL:http://seclists.org/fulldisclosure/2016/Jul/51`
`URL:https://security-tracker.debian.org/tracker/CVE-2016-6210`

... continues on next page ...

... continued from previous page ...

URL:http://openwall.com/lists/oss-security/2016/08/01/2

High (CVSS: 7.5)
NVT: OpenSSH Multiple Vulnerabilities Jan17 (Windows)

**Summary**
This host is installed with openssh and is prone to multiple vulnerabilities.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successfully exploiting this issue allows local users to obtain sensitive private-key information, to gain privileges, conduct a senial-of-service condition and allows remote attackers to execute arbitrary local PKCS#11 modules.
Impact Level: Application

**Solution**
**Solution type:** VendorFix
Upgrade to OpenSSH version 7.4 or later. For updates refer to http://www.openssh.com

**Affected Software/OS**
OpenSSH versions before 7.4 on Windows

**Vulnerability Insight**
Multiple flaws exists due to, - An 'authfile.c' script does not properly consider the effects of realloc on buffer contents. - The shared memory manager (associated with pre-authentication compression) does not ensure that a bounds check is enforced by all compilers. - The sshd in OpenSSH creates forwarded Unix-domain sockets as root, when privilege separation is not used. - An untrusted search path vulnerability in ssh-agent.c in ssh-agent. - NULL pointer dereference error due to an out-of-sequence NEWKEYS message.

**Vulnerability Detection Method**
Get the installed version with the help of detect NVT and check the version is vulnerable or not.
Details: OpenSSH Multiple Vulnerabilities Jan17 (Windows)
OID:1.3.6.1.4.1.25623.1.0.810325
Version used: $Revision: 5084 $

**References**
CVE: CVE-2016-10009, CVE-2016-10010, CVE-2016-10011, CVE-2016-10012, CVE-2016-10
↪708
BID:94968, 94972, 94977, 94975
Other:
URL:https://www.openssh.com/txt/release-7.4
URL:http://www.openwall.com/lists/oss-security/2016/12/19/2
URL:http://blog.swiecki.net/2018/01/fuzzing-tcp-servers.html

... continues on next page ...

... continued from previous page ...

URL:https://anongit.mindrot.org/openssh.git/commit/?id=28652bca29046f62c7045e
↪933e6b931de1d16737

[ return to 127.0.0.10 ]

### 2.7.3 Medium 3389/tcp

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: SSL/TLS: Report Weak Cipher Suites
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: $Revision: 4863 $

**References**
CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000
Other:
URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-
↪1465_update_6.html
URL:https://bettercrypto.org/
URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: `SSL/TLS: Certificate Signed Using A Weak Signature Algorithm`
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: `$Revision: 4781 $`

**References**
```
Other:
  URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with
↪-sha-1-based-signature-algorithms/
```

[ return to 127.0.0.10 ]

### 2.7.4 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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: 4490 $`

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

[ return to 127.0.0.10 ]

### 2.7.5 Medium 135/tcp

Medium (CVSS: 5.0)
NVT: DCE/RPC and MSRPC Services Enumeration Reporting

**Summary**
Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

... continues on next page ...

... continued from previous page ...

**Impact**
An attacker may use this fact to gain more knowledge about the remote host.

**Solution**
**Solution type:** Mitigation
Filter incoming traffic to this ports.

**Vulnerability Detection Method**
Details: `DCE/RPC and MSRPC Services Enumeration Reporting`
OID:1.3.6.1.4.1.25623.1.0.10736
Version used: `$Revision: 4998 $`

[ return to 127.0.0.10 ]

### 2.7.6 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

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

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

... continues on next page ...

. . . continued from previous page . . .

**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: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.10 ]

### 2.7.7 Low 22/tcp

Low (CVSS: 2.6)
NVT: SSH Weak MAC Algorithms Supported

**Summary**
The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Disable the weak MAC algorithms.

**Vulnerability Detection Method**
Details: `SSH Weak MAC Algorithms Supported`
OID:1.3.6.1.4.1.25623.1.0.105610
Version used: `$Revision: 4490 $`

[ return to 127.0.0.10 ]

## 2.8 127.0.0.22

Host scan start Tue Feb 21 15:24:48 2017 UTC
Host scan end Tue Feb 21 15:58:28 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 3389/tcp | High |
| 445/tcp | High |
| 135/tcp | Medium |

### 2.8.1 High 3389/tcp

High (CVSS: 9.3)
NVT: Microsoft Remote Desktop Protocol Remote Code Execution Vulnerabilities (2671387)

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS12-020.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation could allow remote attackers to execute arbitrary code as the logged-on user or cause a denial of service condition.

**Solution**
**Solution type:** VendorFix
Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory from the below link,
http://technet.microsoft.com/en-us/security/bulletin/ms12-020

**Affected Software/OS**
Microsoft Windows 7 Service Pack 1 and prior
Microsoft Windows XP Service Pack 3 and prior
Microsoft Windows 2K3 Service Pack 2 and prior
Microsoft Windows Vista Service Pack 2 and prior
Microsoft Windows Server 2008 Service Pack 2 and prior

**Vulnerability Insight**
The flaws are due to the way Remote Desktop Protocol accesses an object in memory that has been improperly initialized or has been deleted and the way RDP service processes the packets.

**Vulnerability Detection Method**
Details: `Microsoft Remote Desktop Protocol Remote Code Execution Vulnerabilities (267138.`
`↪..`
OID:1.3.6.1.4.1.25623.1.0.902818
Version used: `$Revision: 4234 $`

**References**
```
CVE: CVE-2012-0002, CVE-2012-0152
BID:52353, 52354
Other:
  URL:http://blog.binaryninjas.org/?p=58
   URL:http://secunia.com/advisories/48395
   URL:http://support.microsoft.com/kb/2671387
   URL:http://www.securitytracker.com/id/1026790
   URL:http://technet.microsoft.com/en-us/security/bulletin/ms12-020
```

[ return to 127.0.0.22 ]

### 2.8.2 High 445/tcp

High (CVSS: 0.0)
NVT: SMBv1 enabled (Remote Check)

**Summary**
The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.
This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices.

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

**Log Method**
Details: `SMBv1 enabled (Remote Check)`
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: `$Revision: 5222 $`

**References**
```
Other:
  URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best
↪-Practices
   URL:https://support.microsoft.com/en-us/kb/2696547
   URL:https://support.microsoft.com/en-us/kb/204279
   URL:https://technet.microsoft.com/en-us/library/security/MS17-010
```

High (CVSS: 10.0)
NVT: Microsoft Windows SMB Server NTLM Multiple Vulnerabilities (971468)

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS10-012.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

... continues on next page ...

. . . continued from previous page . . .

**Impact**
Successful exploitation will allow remote attackers to execute arbitrary code or cause a denial of service or bypass the authentication mechanism via brute force technique. Impact Level: System/Application

**Solution**
**Solution type:** VendorFix
Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory from the below link, http://www.microsoft.com/technet/security/bulletin/ms10-012.mspx

**Affected Software/OS**
Microsoft Windows 7 Microsoft Windows 2000 Service Pack and prior Microsoft Windows XP Service Pack 3 and prior Microsoft Windows Vista Service Pack 2 and prior Microsoft Windows Server 2003 Service Pack 2 and prior Microsoft Windows Server 2008 Service Pack 2 and prior

**Vulnerability Insight**
- An input validation error exists while processing SMB requests and can be exploited to cause a buffer overflow via a specially crafted SMB packet. - An error exists in the SMB implementation while parsing SMB packets during the Negotiate phase causing memory corruption via a specially crafted SMB packet. - NULL pointer dereference error exists in SMB while verifying the 'share' and 'servername' fields in SMB packets causing denial of service. - A lack of cryptographic entropy when the SMB server generates challenges during SMB NTLM authentication and can be exploited to bypass the authentication mechanism.

**Vulnerability Detection Method**
Details: `Microsoft Windows SMB Server NTLM Multiple Vulnerabilities (971468)`
OID:1.3.6.1.4.1.25623.1.0.902269
Version used: `$Revision: 5136 $`

**References**
```
CVE: CVE-2010-0020, CVE-2010-0021, CVE-2010-0022, CVE-2010-0231
Other:
  URL:http://secunia.com/advisories/38510/
   URL:http://support.microsoft.com/kb/971468
   URL:http://www.vupen.com/english/advisories/2010/0345
   URL:http://www.microsoft.com/technet/security/bulletin/ms10-012.mspx
```

High (CVSS: 10.0)
NVT: Vulnerabilities in SMB Could Allow Remote Code Execution (958687) - Remote

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS09-001.

**Vulnerability Detection Result**

. . . continues on next page . . .

... continued from previous page ...

Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation could allow remote unauthenticated attackers to cause denying the service by sending a specially crafted network message to a system running the server service.
Impact Level: System/Network

**Solution**
**Solution type:** VendorFix
Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory from the below link, http://www.microsoft.com/technet/security/bulletin/ms09-001.mspx

**Affected Software/OS**
Microsoft Windows 2K Service Pack 4 and prior.
Microsoft Windows XP Service Pack 3 and prior.
Microsoft Windows 2003 Service Pack 2 and prior.

**Vulnerability Insight**
The issue is due to the way Server Message Block (SMB) Protocol software handles specially crafted SMB packets.

**Vulnerability Detection Method**
Details: `Vulnerabilities in SMB Could Allow Remote Code Execution (958687) - Remote`
OID:1.3.6.1.4.1.25623.1.0.900233
Version used: `$Revision: 4692 $`

**References**
```
CVE: CVE-2008-4114, CVE-2008-4834, CVE-2008-4835
BID:31179
Other:
  URL:http://www.milw0rm.com/exploits/6463
   URL:http://www.microsoft.com/technet/security/bulletin/ms09-001.mspx
```

[ return to 127.0.0.22 ]

### 2.8.3 Medium 135/tcp

Medium (CVSS: 5.0)
NVT: DCE/RPC and MSRPC Services Enumeration Reporting

**Summary**
Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries.

... continues on next page ...

. . . continued from previous page . . .

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker may use this fact to gain more knowledge about the remote host.

**Solution**
**Solution type:** Mitigation
Filter incoming traffic to this ports.

**Vulnerability Detection Method**
Details: `DCE/RPC and MSRPC Services Enumeration Reporting`
OID:1.3.6.1.4.1.25623.1.0.10736
Version used: `$Revision: 4998 $`

[ return to 127.0.0.22 ]

## 2.9 127.0.0.44

Host scan start Tue Feb 21 15:24:45 2017 UTC
Host scan end Tue Feb 21 15:43:02 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 22/tcp | High |
| 22/tcp | Medium |
| general/tcp | Low |
| 22/tcp | Low |

### 2.9.1 High 22/tcp

High (CVSS: 10.0)
NVT: Default password ‘WhatsHappeningNow‘ for ‘insight‘ account

**Summary**
The remote device is prone to a default account authentication bypass vulnerability.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
This issue may be exploited by a remote attacker to gain access to sensitive information or modify system configuration.

**Solution**

. . . continues on next page . . .

... continued from previous page ...

**Solution type:** Workaround
Change the password

**Vulnerability Detection Method**
Try to login as ‘insight‘ with password ‘WhatsHappeningNow‘.
Details: `Default password ‘WhatsHappeningNow‘ for ‘insight‘ account`
OID:1.3.6.1.4.1.25623.1.0.140110
Version used: `$Revision: 4868 $`

[ return to 127.0.0.44 ]

### 2.9.2 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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: 4490 $`

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

[ return to 127.0.0.44 ]

### 2.9.3 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.44 ]

### 2.9.4 Low 22/tcp

| Low (CVSS: 2.6)<br>NVT: SSH Weak MAC Algorithms Supported |
|---|
| **Summary**<br>The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms. |
| **Vulnerability Detection Result**<br>Vulnerability was detected according to the Vulnerability Detection Method. |
| **Solution**<br>**Solution type:** Mitigation<br>Disable the weak MAC algorithms. |
| **Vulnerability Detection Method**<br>Details: `SSH Weak MAC Algorithms Supported`<br>OID:1.3.6.1.4.1.25623.1.0.105610<br>Version used: `$Revision: 4490 $` |

[ return to 127.0.0.44 ]

## 2.10 127.0.0.26

Host scan start Tue Feb 21 15:24:49 2017 UTC
Host scan end Tue Feb 21 16:16:03 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 445/tcp | High |
| 3389/tcp | High |
| 135/tcp | Medium |
| 22/tcp | Medium |
| 22/tcp | Low |

### 2.10.1 High 445/tcp

| High (CVSS: 10.0)<br>NVT: Vulnerabilities in SMB Could Allow Remote Code Execution (958687) - Remote |
|---|
| **Summary**<br>This host is missing a critical security update according to Microsoft Bulletin MS09-001. |
| **Vulnerability Detection Result**<br>Vulnerability was detected according to the Vulnerability Detection Method. |
| |
| . . . continues on next page . . . |

... continued from previous page ...

**Impact**
Successful exploitation could allow remote unauthenticated attackers to cause denying the service by sending a specially crafted network message to a system running the server service.
Impact Level: System/Network

**Solution**
**Solution type:** VendorFix
Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory from the below link, http://www.microsoft.com/technet/security/bulletin/ms09-001.mspx

**Affected Software/OS**
Microsoft Windows 2K Service Pack 4 and prior.
Microsoft Windows XP Service Pack 3 and prior.
Microsoft Windows 2003 Service Pack 2 and prior.

**Vulnerability Insight**
The issue is due to the way Server Message Block (SMB) Protocol software handles specially crafted SMB packets.

**Vulnerability Detection Method**
Details: `Vulnerabilities in SMB Could Allow Remote Code Execution (958687) - Remote`
OID:1.3.6.1.4.1.25623.1.0.900233
Version used: `$Revision: 4692 $`

**References**
`CVE: CVE-2008-4114, CVE-2008-4834, CVE-2008-4835`
`BID:31179`
`Other:`
`URL:http://www.milw0rm.com/exploits/6463`
`URL:http://www.microsoft.com/technet/security/bulletin/ms09-001.mspx`

High (CVSS: 0.0)
NVT: SMBv1 enabled (Remote Check)

**Summary**
The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.
This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround

... continues on next page ...

... continued from previous page ...

Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices.

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

**Log Method**
Details: `SMBv1 enabled (Remote Check)`
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: `$Revision: 5222 $`

**References**
```
Other:
  URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best
↪-Practices
   URL:https://support.microsoft.com/en-us/kb/2696547
   URL:https://support.microsoft.com/en-us/kb/204279
   URL:https://technet.microsoft.com/en-us/library/security/MS17-010
```

High (CVSS: 10.0)
NVT: Microsoft Windows SMB Server NTLM Multiple Vulnerabilities (971468)

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS10-012.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow remote attackers to execute arbitrary code or cause a denial of service or bypass the authentication mechanism via brute force technique. Impact Level: System/Application

**Solution**
**Solution type:** VendorFix
Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory from the below link, http://www.microsoft.com/technet/security/bulletin/ms10-012.mspx

**Affected Software/OS**
Microsoft Windows 7 Microsoft Windows 2000 Service Pack and prior Microsoft Windows XP Service Pack 3 and prior Microsoft Windows Vista Service Pack 2 and prior Microsoft Windows Server 2003 Service Pack 2 and prior Microsoft Windows Server 2008 Service Pack 2 and prior

**Vulnerability Insight**

... continues on next page ...

... continued from previous page ...

- An input validation error exists while processing SMB requests and can be exploited to cause a buffer overflow via a specially crafted SMB packet. - An error exists in the SMB implementation while parsing SMB packets during the Negotiate phase causing memory corruption via a specially crafted SMB packet. - NULL pointer dereference error exists in SMB while verifying the 'share' and 'servername' fields in SMB packets causing denial of service. - A lack of cryptographic entropy when the SMB server generates challenges during SMB NTLM authentication and can be exploited to bypass the authentication mechanism.

**Vulnerability Detection Method**
Details: `Microsoft Windows SMB Server NTLM Multiple Vulnerabilities (971468)`
OID:1.3.6.1.4.1.25623.1.0.902269
Version used: `$Revision: 5136 $`

**References**
`CVE: CVE-2010-0020, CVE-2010-0021, CVE-2010-0022, CVE-2010-0231`
`Other:`
`URL:http://secunia.com/advisories/38510/`
`URL:http://support.microsoft.com/kb/971468`
`URL:http://www.vupen.com/english/advisories/2010/0345`
`URL:http://www.microsoft.com/technet/security/bulletin/ms10-012.mspx`

[ return to 127.0.0.26 ]

### 2.10.2 High 3389/tcp

High (CVSS: 9.3)
NVT: Microsoft Remote Desktop Protocol Remote Code Execution Vulnerabilities (2671387)

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS12-020.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation could allow remote attackers to execute arbitrary code as the logged-on user or cause a denial of service condition.

**Solution**
**Solution type:** VendorFix
Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory from the below link,
http://technet.microsoft.com/en-us/security/bulletin/ms12-020

**Affected Software/OS**
Microsoft Windows 7 Service Pack 1 and prior

... continues on next page ...

... continued from previous page ...

Microsoft Windows XP Service Pack 3 and prior
Microsoft Windows 2K3 Service Pack 2 and prior
Microsoft Windows Vista Service Pack 2 and prior
Microsoft Windows Server 2008 Service Pack 2 and prior

**Vulnerability Insight**
The flaws are due to the way Remote Desktop Protocol accesses an object in memory that has been improperly initialized or has been deleted and the way RDP service processes the packets.

**Vulnerability Detection Method**
Details: `Microsoft Remote Desktop Protocol Remote Code Execution Vulnerabilities (267138.`
`↪..`
OID:1.3.6.1.4.1.25623.1.0.902818
Version used: `$Revision: 4234 $`

**References**
`CVE: CVE-2012-0002, CVE-2012-0152`
`BID:52353, 52354`
`Other:`
`URL:http://blog.binaryninjas.org/?p=58`
`URL:http://secunia.com/advisories/48395`
`URL:http://support.microsoft.com/kb/2671387`
`URL:http://www.securitytracker.com/id/1026790`
`URL:http://technet.microsoft.com/en-us/security/bulletin/ms12-020`

[ return to 127.0.0.26 ]

### 2.10.3 Medium 135/tcp

Medium (CVSS: 5.0)
NVT: DCE/RPC and MSRPC Services Enumeration Reporting

**Summary**
Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker may use this fact to gain more knowledge about the remote host.

**Solution**
**Solution type:** Mitigation
Filter incoming traffic to this ports.

... continues on next page ...

... continued from previous page ...

**Vulnerability Detection Method**
Details: `DCE/RPC and MSRPC Services Enumeration Reporting`
OID:1.3.6.1.4.1.25623.1.0.10736
Version used: `$Revision: 4998 $`

[ return to 127.0.0.26 ]

### 2.10.4 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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: 4490 $`

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

[ return to 127.0.0.26 ]

### 2.10.5 Low 22/tcp

| Low (CVSS: 2.6)<br>NVT: SSH Weak MAC Algorithms Supported |
|---|
| **Summary**<br>The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms. |
| **Vulnerability Detection Result**<br>Vulnerability was detected according to the Vulnerability Detection Method. |
| **Solution**<br>**Solution type:** Mitigation<br>Disable the weak MAC algorithms. |
| **Vulnerability Detection Method**<br>Details: `SSH Weak MAC Algorithms Supported`<br>OID:1.3.6.1.4.1.25623.1.0.105610<br>Version used: `$Revision: 4490 $` |

[ return to 127.0.0.26 ]

## 2.11 127.0.0.13

| | |
|---|---|
| Host scan start | Tue Feb 21 15:24:50 2017 UTC |
| Host scan end | Tue Feb 21 16:32:34 2017 UTC |

| Service (Port) | Threat Level |
|---|---|
| 22/tcp | High |
| general/tcp | High |
| 445/tcp | High |
| 22/tcp | Medium |
| 2011/tcp | Medium |
| 22/tcp | Low |

### 2.11.1 High 22/tcp

| High (CVSS: 7.8)<br>NVT: OpenSSH Denial of Service And User Enumeration Vulnerabilities (Windows) |
|---|
| **Summary**<br>This host is installed with openssh and is prone to denial of service and user enumeration vulnerabilities. |
| **Vulnerability Detection Result** |
| . . . continues on next page . . . |

... continued from previous page ...

Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successfully exploiting this issue allows remote attackers to cause a denial of service (crypt CPU consumption) and to enumerate users by leveraging the timing difference between responses when a large password is provided.

**Solution**
**Solution type:** VendorFix
Upgrade to OpenSSH version 7.3 or later. For updates refer to http://www.openssh.com

**Affected Software/OS**
OpenSSH versions before 7.3 on Windows

**Vulnerability Insight**
Multiple flaws exists due to,
- The auth_password function in 'auth-passwd.c' script does not limit password lengths for password authentication.
- The sshd in OpenSSH, when SHA256 or SHA512 are used for user password hashing uses BLOWFISH hashing on a static password when the username does not exist and it takes much longer to calculate SHA256/SHA512 hash than BLOWFISH hash.

**Vulnerability Detection Method**
Checks if a vulnerable version is present on the target host.
Details: `OpenSSH Denial of Service And User Enumeration Vulnerabilities (Windows)`
OID:1.3.6.1.4.1.25623.1.0.809121
Version used: `$Revision: 5083 $`

**References**
`CVE: CVE-2016-6515, CVE-2016-6210`
`BID:92212`
`Other:`
`URL:http://www.openssh.com/txt/release-7.3`
`URL:http://seclists.org/fulldisclosure/2016/Jul/51`
`URL:https://security-tracker.debian.org/tracker/CVE-2016-6210`
`URL:http://openwall.com/lists/oss-security/2016/08/01/2`

High (CVSS: 7.5)
NVT: OpenSSH Multiple Vulnerabilities Jan17 (Windows)

**Summary**
This host is installed with openssh and is prone to multiple vulnerabilities.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

... continues on next page ...

... continued from previous page ...

**Impact**
Successfully exploiting this issue allows local users to obtain sensitive private-key information, to gain privileges, conduct a senial-of-service condition and allows remote attackers to execute arbitrary local PKCS#11 modules.
Impact Level: Application

**Solution**
**Solution type:** VendorFix
Upgrade to OpenSSH version 7.4 or later. For updates refer to http://www.openssh.com

**Affected Software/OS**
OpenSSH versions before 7.4 on Windows

**Vulnerability Insight**
Multiple flaws exists due to, - An 'authfile.c' script does not properly consider the effects of realloc on buffer contents. - The shared memory manager (associated with pre-authentication compression) does not ensure that a bounds check is enforced by all compilers. - The sshd in OpenSSH creates forwarded Unix-domain sockets as root, when privilege separation is not used. - An untrusted search path vulnerability in ssh-agent.c in ssh-agent. - NULL pointer dereference error due to an out-of-sequence NEWKEYS message.

**Vulnerability Detection Method**
Get the installed version with the help of detect NVT and check the version is vulnerable or not.
Details: `OpenSSH Multiple Vulnerabilities Jan17 (Windows)`
OID:1.3.6.1.4.1.25623.1.0.810325
Version used: `$Revision: 5084 $`

**References**
```
CVE: CVE-2016-10009, CVE-2016-10010, CVE-2016-10011, CVE-2016-10012, CVE-2016-10
↪708
BID:94968, 94972, 94977, 94975
Other:
  URL:https://www.openssh.com/txt/release-7.4
   URL:http://www.openwall.com/lists/oss-security/2016/12/19/2
   URL:http://blog.swiecki.net/2018/01/fuzzing-tcp-servers.html
   URL:https://anongit.mindrot.org/openssh.git/commit/?id=28652bca29046f62c7045e
↪933e6b931de1d16737
```

[ return to 127.0.0.13 ]

### 2.11.2 High general/tcp

High (CVSS: 10.0)
NVT: OS End Of Life Detection

**Summary**

... continues on next page ...

... continued from previous page ...

OS End Of Life Detection
The Operating System on the remote host has reached the end of life and should not be used anymore.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation

**Vulnerability Detection Method**
Details: `OS End Of Life Detection`
OID:1.3.6.1.4.1.25623.1.0.103674
Version used: `$Revision: 4111 $`

[ return to 127.0.0.13 ]

### 2.11.3 High 445/tcp

High (CVSS: 0.0)
NVT: SMBv1 enabled (Remote Check)

**Summary**
The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.
This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices.

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

**Log Method**
Details: `SMBv1 enabled (Remote Check)`
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: `$Revision: 5222 $`

**References**

... continues on next page ...

... continued from previous page ...

```
Other:
  URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best
↪-Practices
   URL:https://support.microsoft.com/en-us/kb/2696547
   URL:https://support.microsoft.com/en-us/kb/204279
   URL:https://technet.microsoft.com/en-us/library/security/MS17-010
```

[ return to 127.0.0.13 ]

### 2.11.4 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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: 4490 $`

**References**

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

[ return to 127.0.0.13 ]

### 2.11.5 Medium 2011/tcp

Medium (CVSS: 6.8)
NVT: SSL/TLS: OpenSSL CCS Man in the Middle Security Bypass Vulnerability

**Summary**
OpenSSL is prone to security-bypass vulnerability.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successfully exploiting this issue may allow attackers to obtain sensitive information by conducting a man-in-the-middle attack. This may lead to other attacks.

**Solution**
**Solution type:** VendorFix
Updates are available.

**Affected Software/OS**
OpenSSL before 0.9.8za, 1.0.0 before 1.0.0m and 1.0.1 before 1.0.1h

**Vulnerability Insight**
OpenSSL does not properly restrict processing of ChangeCipherSpec messages, which allows man-in-the-middle attackers to trigger use of a zero-length master key in certain OpenSSL-to-OpenSSL communications, and consequently hijack sessions or obtain sensitive information, via a crafted TLS handshake, aka the 'CCS Injection' vulnerability.

**Vulnerability Detection Method**
Send two SSL ChangeCipherSpec request and check the response.
Details: `SSL/TLS: OpenSSL CCS Man in the Middle Security Bypass Vulnerability`
OID:1.3.6.1.4.1.25623.1.0.105042
Version used: `$Revision: 4679 $`

**References**
`CVE: CVE-2014-0224`
`BID:67899`
`Other:`
`URL:http://www.securityfocus.com/bid/67899`
`URL:http://openssl.org/`

Medium (CVSS: 5.0)
NVT: SSL/TLS: Certificate Expired

**Summary**
The remote server's SSL/TLS certificate has already expired.

. . . continues on next page . . .

... continued from previous page ...

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Replace the SSL/TLS certificate by a new one.

**Vulnerability Insight**
This script checks expiry dates of certificates associated with SSL/TLS-enabled services on the target and reports whether any have already expired.

**Vulnerability Detection Method**
Details: `SSL/TLS: Certificate Expired`
OID:1.3.6.1.4.1.25623.1.0.103955
Version used: `$Revision: 4765 $`

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Weak Cipher Suites`

... continues on next page ...

... continued from previous page ...

OID:1.3.6.1.4.1.25623.1.0.103440
Version used: $Revision: 4863 $

**References**
CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000
Other:
URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html
URL:https://bettercrypto.org/
URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/

Medium (CVSS: 4.3)
NVT: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection

**Summary**
It was possible to detect the usage of the deprecated SSLv2 and/or SSLv3 protocol on this system.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to use the known cryptographic flaws to eavesdrop the connection between clients and the service to get access to sensitive data transferred within the secured connection.

**Solution**
**Solution type:** Mitigation
It is recommended to disable the deprecated SSLv2 and/or SSLv3 protocols in favor of the TLSv1+ protocols. Please see the references for more information.

**Affected Software/OS**
All services providing an encrypted communication using the SSLv2 and/or SSLv3 protocols.

**Vulnerability Insight**
The SSLv2 and SSLv3 protocols containing known cryptographic flaws like:
- Padding Oracle On Downgraded Legacy Encryption (POODLE, CVE-2014-3566)
- Decrypting RSA with Obsolete and Weakened eNcryption (DROWN, CVE-2016-0800)

**Vulnerability Detection Method**
Check the used protocols of the services provided by this system.
Details: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection
OID:1.3.6.1.4.1.25623.1.0.111012
Version used: $Revision: 4686 $

**References**

... continues on next page ...

... continued from previous page ...

CVE: CVE-2016-0800, CVE-2014-3566
Other:
URL:https://www.enisa.europa.eu/activities/identity-and-trust/library/deliverables/algorithms-key-sizes-and-parameters-report
URL:https://bettercrypto.org/
URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
URL:https://drownattack.com/
URL:https://www.imperialviolet.org/2014/10/14/poodle.html

Medium (CVSS: 4.3)
NVT: SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability (POODLE)

**Summary**
This host is prone to an information disclosure vulnerability.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow a man-in-the-middle attackers gain access to the plain text data stream.

**Solution**
**Solution type:** Mitigation
Possible Mitigations are:
- Disable SSLv3
- Disable cipher suites supporting CBC cipher modes
- Enable TLS_FALLBACK_SCSV if the service is providing TLSv1.0+

**Vulnerability Insight**
The flaw is due to the block cipher padding not being deterministic and not covered by the Message Authentication Code

**Vulnerability Detection Method**
Evaluate previous collected information about this service.
Details: SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability ...
OID:1.3.6.1.4.1.25623.1.0.802087
Version used: $Revision: 4749 $

**References**
CVE: CVE-2014-3566
BID:70574
Other:
URL:https://www.openssl.org/~bodo/ssl-poodle.pdf
URL:https://www.imperialviolet.org/2014/10/14/poodle.html

... continues on next page ...

... continued from previous page ...

URL:https://www.dfranke.us/posts/2014-10-14-how-poodle-happened.html
URL:http://googleonlinesecurity.blogspot.in/2014/10/this-poodle-bites-exploit
↪ing-ssl-30.html

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: $Revision: 4781 $

**References**
Other:
URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with
↪-sha-1-based-signature-algorithms/

[ return to 127.0.0.13 ]

### 2.11.6 Low 22/tcp

| Low (CVSS: 2.6)<br>NVT: SSH Weak MAC Algorithms Supported |
|---|
| **Summary**<br>The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms. |
| **Vulnerability Detection Result**<br>Vulnerability was detected according to the Vulnerability Detection Method. |
| **Solution**<br>**Solution type:** Mitigation<br>Disable the weak MAC algorithms. |
| **Vulnerability Detection Method**<br>Details: `SSH Weak MAC Algorithms Supported`<br>OID:1.3.6.1.4.1.25623.1.0.105610<br>Version used: `$Revision: 4490 $` |

[ return to 127.0.0.13 ]

## 2.12 127.0.0.7

Host scan start Tue Feb 21 15:24:48 2017 UTC
Host scan end Tue Feb 21 15:57:37 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 3389/tcp | High |
| 445/tcp | High |
| 135/tcp | Medium |
| 8080/tcp | Medium |
| 443/tcp | Medium |
| 8098/tcp | Medium |
| 80/tcp | Medium |
| 21/tcp | Medium |

### 2.12.1 High 3389/tcp

| High (CVSS: 9.3)<br>NVT: Microsoft Remote Desktop Protocol Remote Code Execution Vulnerabilities (2671387) |
|---|
| |
| . . . continues on next page . . . |

... continued from previous page ...

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS12-020.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation could allow remote attackers to execute arbitrary code as the logged-on user or cause a denial of service condition.

**Solution**
**Solution type:** VendorFix
Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory from the below link,
http://technet.microsoft.com/en-us/security/bulletin/ms12-020

**Affected Software/OS**
Microsoft Windows 7 Service Pack 1 and prior
Microsoft Windows XP Service Pack 3 and prior
Microsoft Windows 2K3 Service Pack 2 and prior
Microsoft Windows Vista Service Pack 2 and prior
Microsoft Windows Server 2008 Service Pack 2 and prior

**Vulnerability Insight**
The flaws are due to the way Remote Desktop Protocol accesses an object in memory that has been improperly initialized or has been deleted and the way RDP service processes the packets.

**Vulnerability Detection Method**
Details: `Microsoft Remote Desktop Protocol Remote Code Execution Vulnerabilities (267138.`
`↪..`
OID:1.3.6.1.4.1.25623.1.0.902818
Version used: `$Revision: 4234 $`

**References**
`CVE: CVE-2012-0002, CVE-2012-0152`
`BID:52353, 52354`
`Other:`
`URL:http://blog.binaryninjas.org/?p=58`
`URL:http://secunia.com/advisories/48395`
`URL:http://support.microsoft.com/kb/2671387`
`URL:http://www.securitytracker.com/id/1026790`
`URL:http://technet.microsoft.com/en-us/security/bulletin/ms12-020`

[ return to 127.0.0.7 ]

### 2.12.2 High 445/tcp

High (CVSS: 10.0)
NVT: Vulnerabilities in SMB Could Allow Remote Code Execution (958687) - Remote

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS09-001.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation could allow remote unauthenticated attackers to cause denying the service by sending a specially crafted network message to a system running the server service.
Impact Level: System/Network

**Solution**
**Solution type:** VendorFix
Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory from the below link, http://www.microsoft.com/technet/security/bulletin/ms09-001.mspx

**Affected Software/OS**
Microsoft Windows 2K Service Pack 4 and prior.
Microsoft Windows XP Service Pack 3 and prior.
Microsoft Windows 2003 Service Pack 2 and prior.

**Vulnerability Insight**
The issue is due to the way Server Message Block (SMB) Protocol software handles specially crafted SMB packets.

**Vulnerability Detection Method**
Details: `Vulnerabilities in SMB Could Allow Remote Code Execution (958687) - Remote`
OID:1.3.6.1.4.1.25623.1.0.900233
Version used: `$Revision: 4692 $`

**References**
```
CVE: CVE-2008-4114, CVE-2008-4834, CVE-2008-4835
BID:31179
Other:
  URL:http://www.milw0rm.com/exploits/6463
   URL:http://www.microsoft.com/technet/security/bulletin/ms09-001.mspx
```

High (CVSS: 0.0)
NVT: SMBv1 enabled (Remote Check)

**Summary**
The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.

. . . continues on next page . . .

... continued from previous page ...

This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices.

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

**Log Method**
Details: `SMBv1 enabled (Remote Check)`
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: `$Revision: 5222 $`

**References**
```
Other:
  URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best
↪-Practices
   URL:https://support.microsoft.com/en-us/kb/2696547
   URL:https://support.microsoft.com/en-us/kb/204279
   URL:https://technet.microsoft.com/en-us/library/security/MS17-010
```

High (CVSS: 10.0)
NVT: Microsoft Windows SMB Server NTLM Multiple Vulnerabilities (971468)

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS10-012.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow remote attackers to execute arbitrary code or cause a denial of service or bypass the authentication mechanism via brute force technique. Impact Level: System/Application

**Solution**
**Solution type:** VendorFix

... continues on next page ...

... continued from previous page ...

Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory from the below link, http://www.microsoft.com/technet/security/bulletin/ms10-012.mspx

**Affected Software/OS**
Microsoft Windows 7 Microsoft Windows 2000 Service Pack and prior Microsoft Windows XP Service Pack 3 and prior Microsoft Windows Vista Service Pack 2 and prior Microsoft Windows Server 2003 Service Pack 2 and prior Microsoft Windows Server 2008 Service Pack 2 and prior

**Vulnerability Insight**
- An input validation error exists while processing SMB requests and can be exploited to cause a buffer overflow via a specially crafted SMB packet. - An error exists in the SMB implementation while parsing SMB packets during the Negotiate phase causing memory corruption via a specially crafted SMB packet. - NULL pointer dereference error exists in SMB while verifying the 'share' and 'servername' fields in SMB packets causing denial of service. - A lack of cryptographic entropy when the SMB server generates challenges during SMB NTLM authentication and can be exploited to bypass the authentication mechanism.

**Vulnerability Detection Method**
Details: `Microsoft Windows SMB Server NTLM Multiple Vulnerabilities (971468)`
OID:1.3.6.1.4.1.25623.1.0.902269
Version used: `$Revision: 5136 $`

**References**
```
CVE: CVE-2010-0020, CVE-2010-0021, CVE-2010-0022, CVE-2010-0231
Other:
  URL:http://secunia.com/advisories/38510/
   URL:http://support.microsoft.com/kb/971468
   URL:http://www.vupen.com/english/advisories/2010/0345
   URL:http://www.microsoft.com/technet/security/bulletin/ms10-012.mspx
```

[ return to 127.0.0.7 ]

### 2.12.3 Medium 135/tcp

Medium (CVSS: 5.0)
NVT: DCE/RPC and MSRPC Services Enumeration Reporting

**Summary**
Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

... continues on next page ...

. . . continued from previous page . . .

**Impact**
An attacker may use this fact to gain more knowledge about the remote host.

**Solution**
**Solution type:** Mitigation
Filter incoming traffic to this ports.

**Vulnerability Detection Method**
Details: `DCE/RPC and MSRPC Services Enumeration Reporting`
OID:1.3.6.1.4.1.25623.1.0.10736
Version used: `$Revision: 4998 $`

[ return to 127.0.0.7 ]

### 2.12.4 Medium 8080/tcp

Medium (CVSS: 5.0)
NVT: IIS Service Pack - 404

**Summary**
Ensure that the server is running the latest stable Service Pack

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** VendorFix
The Patch level (Service Pack) of the remote IIS server appears to be lower than the current IIS service pack level. As each service pack typically contains many security patches, the server may be at risk.
Caveat: This test makes assumptions of the remote patch level based on static return values (Content-Length) within the IIS Servers 404 error message. As such, the test can not be totally reliable and should be manually confirmed.

**Vulnerability Detection Method**
Details: `IIS Service Pack - 404`
OID:1.3.6.1.4.1.25623.1.0.11874
Version used: `$Revision: 4703 $`

[ return to 127.0.0.7 ]

### 2.12.5 Medium 443/tcp

Medium (CVSS: 5.0)
NVT: Microsoft IIS Tilde Character Information Disclosure Vulnerability

**Summary**
This host is running Microsoft IIS Webserver and is prone to information disclosure vulnerability.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow remote attackers to obtain sensitive information that could aid in further attacks.

**Solution**
**Solution type:** WillNotFix
No known solution was made available for at least one year since the disclosure of this vulnerability. Likely none will be provided anymore. General solution options are to upgrade to a newer release, disable respective features, remove the product or replace the product by another one.

**Affected Software/OS**
Microsoft Internet Information Services versions 7.5 and prior

**Vulnerability Insight**
Microsoft IIS fails to validate a specially crafted GET request containing a ' ' tilde character, which allows to disclose all short-names of folders and files having 4 letters extensions.

**Vulnerability Detection Method**
Details: `Microsoft IIS Tilde Character Information Disclosure Vulnerability`
OID:1.3.6.1.4.1.25623.1.0.802887
Version used: `$Revision: 3565 $`

**References**
```
BID:54251
Other:
  URL:http://www.exploit-db.com/exploits/19525
   URL:http://code.google.com/p/iis-shortname-scanner-poc
   URL:http://soroush.secproject.com/downloadable/iis_tilde_shortname_disclosure
↪.txt
   URL:http://soroush.secproject.com/downloadable/microsoft_iis_tilde_character_
↪vulnerability_feature.pdf
```

Medium (CVSS: 5.0)
NVT: IIS Service Pack - 404

**Summary**
Ensure that the server is running the latest stable Service Pack

. . . continues on next page . . .

... continued from previous page ...

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** VendorFix
The Patch level (Service Pack) of the remote IIS server appears to be lower than the current IIS service pack level. As each service pack typically contains many security patches, the server may be at risk.
Caveat: This test makes assumptions of the remote patch level based on static return values (Content-Length) within the IIS Servers 404 error message. As such, the test can not be totally reliable and should be manually confirmed.

**Vulnerability Detection Method**
Details: `IIS Service Pack - 404`
OID:1.3.6.1.4.1.25623.1.0.11874
Version used: `$Revision: 4703 $`

[ return to 127.0.0.7 ]

### 2.12.6 Medium 8098/tcp

Medium (CVSS: 5.0)
NVT: SSL/TLS: Report Vulnerable Cipher Suites for HTTPS

**Summary**
This routine reports all SSL/TLS cipher suites accepted by a service where attack vectors exists only on HTTPS services.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Affected Software/OS**
Services accepting vulnerable SSL/TLS cipher suites via HTTPS.

**Vulnerability Insight**
These rules are applied for the evaluation of the vulnerable cipher suites:
- 64-bit block cipher 3DES vulnerable to the SWEET32 attack (CVE-2016-2183).

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Vulnerable Cipher Suites for HTTPS`

... continues on next page ...

... continued from previous page ...

OID:1.3.6.1.4.1.25623.1.0.108031
Version used: $Revision: 5232 $

**References**
CVE: CVE-2016-2183, CVE-2016-6329
Other:
URL:https://bettercrypto.org/
URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
URL:https://sweet32.info/

Medium (CVSS: 5.0)
NVT: SSL/TLS: Certificate Expired

**Summary**
The remote server's SSL/TLS certificate has already expired.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Replace the SSL/TLS certificate by a new one.

**Vulnerability Insight**
This script checks expiry dates of certificates associated with SSL/TLS-enabled services on the target and reports whether any have already expired.

**Vulnerability Detection Method**
Details: SSL/TLS: Certificate Expired
OID:1.3.6.1.4.1.25623.1.0.103955
Version used: $Revision: 4765 $

Medium (CVSS: 4.3)
NVT: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection

**Summary**
It was possible to detect the usage of the deprecated SSLv2 and/or SSLv3 protocol on this system.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**

... continues on next page ...

... continued from previous page ...

An attacker might be able to use the known cryptographic flaws to eavesdrop the connection between clients and the service to get access to sensitive data transferred within the secured connection.

**Solution**
**Solution type:** Mitigation
It is recommended to disable the deprecated SSLv2 and/or SSLv3 protocols in favor of the TLSv1+ protocols. Please see the references for more information.

**Affected Software/OS**
All services providing an encrypted communication using the SSLv2 and/or SSLv3 protocols.

**Vulnerability Insight**
The SSLv2 and SSLv3 protocols containing known cryptographic flaws like:
- Padding Oracle On Downgraded Legacy Encryption (POODLE, CVE-2014-3566)
- Decrypting RSA with Obsolete and Weakened eNcryption (DROWN, CVE-2016-0800)

**Vulnerability Detection Method**
Check the used protocols of the services provided by this system.
Details: `SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection`
OID:1.3.6.1.4.1.25623.1.0.111012
Version used: `$Revision: 4686 $`

**References**
```
CVE: CVE-2016-0800, CVE-2014-3566
Other:
  URL:https://www.enisa.europa.eu/activities/identity-and-trust/library/delivera
↪bles/algorithms-key-sizes-and-parameters-report
   URL:https://bettercrypto.org/
   URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
   URL:https://drownattack.com/
   URL:https://www.imperialviolet.org/2014/10/14/poodle.html
```

## Medium (CVSS: 4.3)
## NVT: SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability (POODLE)

**Summary**
This host is prone to an information disclosure vulnerability.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow a man-in-the-middle attackers gain access to the plain text data stream.

... continues on next page ...

... continued from previous page ...

**Solution**
**Solution type:** Mitigation
Possible Mitigations are:
- Disable SSLv3
- Disable cipher suites supporting CBC cipher modes
- Enable TLS_FALLBACK_SCSV if the service is providing TLSv1.0+

**Vulnerability Insight**
The flaw is due to the block cipher padding not being deterministic and not covered by the Message Authentication Code

**Vulnerability Detection Method**
Evaluate previous collected information about this service.
Details: `SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability ..`
OID:1.3.6.1.4.1.25623.1.0.802087
Version used: `$Revision: 4749 $`

**References**
`CVE: CVE-2014-3566`
`BID:70574`
`Other:`
`URL:https://www.openssl.org/~bodo/ssl-poodle.pdf`
`URL:https://www.imperialviolet.org/2014/10/14/poodle.html`
`URL:https://www.dfranke.us/posts/2014-10-14-how-poodle-happened.html`
`URL:http://googleonlinesecurity.blogspot.in/2014/10/this-poodle-bites-exploiting-ssl-30.html`

Medium (CVSS: 4.3)
NVT: SSL/TLS: RSA Temporary Key Handling 'RSA_EXPORT' Downgrade Issue (FREAK)

**Summary**
This host is accepting 'RSA_EXPORT' cipher suites and is prone to man in the middle attack.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow remote attacker to downgrade the security of a session to use 'RSA_EXPORT' cipher suites, which are significantly weaker than non-export cipher suites. This may allow a man-in-the-middle attacker to more easily break the encryption and monitor or tamper with the encrypted stream.

**Solution**
**Solution type:** VendorFix
- Remove support for 'RSA_EXPORT' cipher suites from the service.

... continues on next page ...

... continued from previous page ...

- If running OpenSSL update to version 0.9.8zd or 1.0.0p or 1.0.1k or later For updates refer to https://www.openssl.org

**Affected Software/OS**
- Hosts accepting 'RSA_EXPORT' cipher suites
- OpenSSL version before 0.9.8zd, 1.0.0 before 1.0.0p, and 1.0.1 before 1.0.1k.

**Vulnerability Insight**
Flaw is due to improper handling RSA temporary keys in a non-export RSA key exchange cipher suite.

**Vulnerability Detection Method**
Check previous collected cipher suites saved in the KB.
Details: `SSL/TLS: RSA Temporary Key Handling 'RSA_EXPORT' Downgrade Issue (FREAK)`
OID:1.3.6.1.4.1.25623.1.0.805142
Version used: `$Revision: 4781 $`

**References**
```
CVE: CVE-2015-0204
BID:71936
Other:
  URL:https://freakattack.com
   URL:http://secpod.org/blog/?p=3818
   URL:http://blog.cryptographyengineering.com/2015/03/attack-of-week-freak-or-f
↪actoring-nsa.html
```

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:

... continues on next page ...

... continued from previous page ...

- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Weak Cipher Suites`
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: `$Revision: 4863 $`

**References**
`CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000`
`Other:`
`URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html`
`URL:https://bettercrypto.org/`
`URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/`

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.

... continues on next page ...

... continued from previous page ...

NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: `SSL/TLS: Certificate Signed Using A Weak Signature Algorithm`
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: `$Revision: 4781 $`

**References**
```
Other:
  URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with
↪-sha-1-based-signature-algorithms/
```

[ return to 127.0.0.7 ]

### 2.12.7 Medium 80/tcp

Medium (CVSS: 5.0)
NVT: Microsoft IIS Tilde Character Information Disclosure Vulnerability

**Summary**
This host is running Microsoft IIS Webserver and is prone to information disclosure vulnerability.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow remote attackers to obtain sensitive information that could aid in further attacks.

**Solution**
**Solution type:** WillNotFix
No known solution was made available for at least one year since the disclosure of this vulnerability. Likely none will be provided anymore. General solution options are to upgrade to a newer release, disable respective features, remove the product or replace the product by another one.

**Affected Software/OS**
Microsoft Internet Information Services versions 7.5 and prior

**Vulnerability Insight**

... continues on next page ...

... continued from previous page ...

Microsoft IIS fails to validate a specially crafted GET request containing a ' ' tilde character, which allows to disclose all short-names of folders and files having 4 letters extensions.

**Vulnerability Detection Method**
Details: `Microsoft IIS Tilde Character Information Disclosure Vulnerability`
OID:1.3.6.1.4.1.25623.1.0.802887
Version used: `$Revision: 3565 $`

**References**
```
BID:54251
Other:
  URL:http://www.exploit-db.com/exploits/19525
   URL:http://code.google.com/p/iis-shortname-scanner-poc
   URL:http://soroush.secproject.com/downloadable/iis_tilde_shortname_disclosure
↪.txt
   URL:http://soroush.secproject.com/downloadable/microsoft_iis_tilde_character_
↪vulnerability_feature.pdf
```

Medium (CVSS: 5.0)
NVT: IIS Service Pack - 404

**Summary**
Ensure that the server is running the latest stable Service Pack

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** VendorFix
The Patch level (Service Pack) of the remote IIS server appears to be lower than the current IIS service pack level. As each service pack typically contains many security patches, the server may be at risk.
Caveat: This test makes assumptions of the remote patch level based on static return values (Content-Length) within the IIS Servers 404 error message. As such, the test can not be totally reliable and should be manually confirmed.

**Vulnerability Detection Method**
Details: `IIS Service Pack - 404`
OID:1.3.6.1.4.1.25623.1.0.11874
Version used: `$Revision: 4703 $`

[ return to 127.0.0.7 ]

### 2.12.8 Medium 21/tcp

Medium (CVSS: 6.4)
NVT: Check for Anonymous FTP Login

**Summary**
This FTP Server allows anonymous logins.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Based on the files accessible via this anonymous FTP login and the permissions of this account an attacker might be able to:
- gain access to sensitive files
- upload or delete files

**Solution**
**Solution type:** Mitigation
If you do not want to share files, you should disable anonymous logins.

**Vulnerability Insight**
A host that provides an FTP service may additionally provide Anonymous FTP access as well. Under this arrangement, users do not strictly need an account on the host. Instead the user typically enters 'anonymous' or 'ftp' when prompted for username. Although users are commonly asked to send their email address as their password, little to no verification is actually performed on the supplied data.

**Vulnerability Detection Method**
Try to login with an anonymous account at the remove FTP service.
Details: `Check for Anonymous FTP Login`
OID:1.3.6.1.4.1.25623.1.0.900600
Version used: `$Revision: 4987 $`

**References**
`Other:`
`URL:https://web.nvd.nist.gov/view/vuln/detail?vulnId=CVE-1999-0497`

[ return to 127.0.0.7 ]

## 2.13 127.0.0.20

| | |
|---|---|
| Host scan start | Tue Feb 21 15:24:57 2017 UTC |
| Host scan end | Tue Feb 21 15:56:50 2017 UTC |

| Service (Port) | Threat Level |
|---|---|
| 445/tcp | High |
| 3389/tcp | Medium |
| 135/tcp | Medium |

. . . (continues) . . .

... (continued) ...

| Service (Port) | Threat Level |
|---|---|
| 8080/tcp | Medium |
| 443/tcp | Medium |
| general/tcp | Low |

### 2.13.1 High 445/tcp

High (CVSS: 0.0)
NVT: SMBv1 enabled (Remote Check)

**Summary**
The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.
This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices.

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

**Log Method**
Details: `SMBv1 enabled (Remote Check)`
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: `$Revision: 5222 $`

**References**
```
Other:
  URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best
↪-Practices
   URL:https://support.microsoft.com/en-us/kb/2696547
   URL:https://support.microsoft.com/en-us/kb/204279
   URL:https://technet.microsoft.com/en-us/library/security/MS17-010
```

[ return to 127.0.0.20 ]

### 2.13.2 Medium 3389/tcp

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Weak Cipher Suites`
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: `$Revision: 4863 $`

**References**
```
CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000
Other:
  URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-
↪1465_update_6.html
   URL:https://bettercrypto.org/
   URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
```

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

. . . continues on next page . . .

... continued from previous page ...

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: `SSL/TLS: Certificate Signed Using A Weak Signature Algorithm`
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: `$Revision: 4781 $`

**References**
```
Other:
  URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with
↪-sha-1-based-signature-algorithms/
```

[ return to 127.0.0.20 ]

### 2.13.3 Medium 135/tcp

Medium (CVSS: 5.0)
NVT: DCE/RPC and MSRPC Services Enumeration Reporting

**Summary**

... continues on next page ...

... continued from previous page ...

Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker may use this fact to gain more knowledge about the remote host.

**Solution**
**Solution type:** Mitigation
Filter incoming traffic to this ports.

**Vulnerability Detection Method**
Details: `DCE/RPC and MSRPC Services Enumeration Reporting`
OID:1.3.6.1.4.1.25623.1.0.10736
Version used: `$Revision: 4998 $`

[ return to 127.0.0.20 ]

### 2.13.4 Medium 8080/tcp

Medium (CVSS: 5.0)
NVT: Missing 'httpOnly' Cookie Attribute

**Summary**
The application is missing the 'httpOnly' cookie attribute

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Set the 'httpOnly' attribute for any session cookie.

**Affected Software/OS**
Application with session handling in cookies.

**Vulnerability Insight**
The flaw is due to a cookie is not using the 'httpOnly' attribute. This allows a cookie to be accessed by JavaScript which could lead to session hijacking attacks.

**Vulnerability Detection Method**
Check all cookies sent by the application for a missing 'httpOnly' attribute
Details: `Missing 'httpOnly' Cookie Attribute`

... continues on next page ...

... continued from previous page ...

OID:1.3.6.1.4.1.25623.1.0.105925
Version used: $Revision: 5270 $

**References**
Other:
URL:https://www.owasp.org/index.php/HttpOnly
URL:https://www.owasp.org/index.php/Testing_for_cookies_attributes_(OTG-SESS-002)

[ return to 127.0.0.20 ]

### 2.13.5 Medium 443/tcp

Medium (CVSS: 5.0)
NVT: SSL/TLS: Report Vulnerable Cipher Suites for HTTPS

**Summary**
This routine reports all SSL/TLS cipher suites accepted by a service where attack vectors exists only on HTTPS services.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Affected Software/OS**
Services accepting vulnerable SSL/TLS cipher suites via HTTPS.

**Vulnerability Insight**
These rules are applied for the evaluation of the vulnerable cipher suites:
- 64-bit block cipher 3DES vulnerable to the SWEET32 attack (CVE-2016-2183).

**Vulnerability Detection Method**
Details: SSL/TLS: Report Vulnerable Cipher Suites for HTTPS
OID:1.3.6.1.4.1.25623.1.0.108031
Version used: $Revision: 5232 $

**References**
CVE: CVE-2016-2183, CVE-2016-6329
Other:
URL:https://bettercrypto.org/
URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/

... continues on next page ...

... continued from previous page ...

URL:https://sweet32.info/

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: SSL/TLS: Report Weak Cipher Suites
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: $Revision: 4863 $

**References**
CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000
Other:
URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html
URL:https://bettercrypto.org/
URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/

Medium (CVSS: 4.3)
NVT: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection

... continues on next page ...

... continued from previous page ...

**Summary**
It was possible to detect the usage of the deprecated SSLv2 and/or SSLv3 protocol on this system.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to use the known cryptographic flaws to eavesdrop the connection between clients and the service to get access to sensitive data transferred within the secured connection.

**Solution**
**Solution type:** Mitigation
It is recommended to disable the deprecated SSLv2 and/or SSLv3 protocols in favor of the TLSv1+ protocols. Please see the references for more information.

**Affected Software/OS**
All services providing an encrypted communication using the SSLv2 and/or SSLv3 protocols.

**Vulnerability Insight**
The SSLv2 and SSLv3 protocols containing known cryptographic flaws like:
- Padding Oracle On Downgraded Legacy Encryption (POODLE, CVE-2014-3566)
- Decrypting RSA with Obsolete and Weakened eNcryption (DROWN, CVE-2016-0800)

**Vulnerability Detection Method**
Check the used protocols of the services provided by this system.
Details: `SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection`
OID:1.3.6.1.4.1.25623.1.0.111012
Version used: `$Revision: 4686 $`

**References**
```
CVE: CVE-2016-0800, CVE-2014-3566
Other:
  URL:https://www.enisa.europa.eu/activities/identity-and-trust/library/delivera
↪bles/algorithms-key-sizes-and-parameters-report
   URL:https://bettercrypto.org/
   URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
   URL:https://drownattack.com/
   URL:https://www.imperialviolet.org/2014/10/14/poodle.html
```

Medium (CVSS: 4.3)
NVT: SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability (POODLE)

**Summary**

... continues on next page ...

... continued from previous page ...

This host is prone to an information disclosure vulnerability.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow a man-in-the-middle attackers gain access to the plain text data stream.

**Solution**
**Solution type:** Mitigation
Possible Mitigations are:
- Disable SSLv3
- Disable cipher suites supporting CBC cipher modes
- Enable TLS_FALLBACK_SCSV if the service is providing TLSv1.0+

**Vulnerability Insight**
The flaw is due to the block cipher padding not being deterministic and not covered by the Message Authentication Code

**Vulnerability Detection Method**
Evaluate previous collected information about this service.
Details: `SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability ..`
OID:1.3.6.1.4.1.25623.1.0.802087
Version used: `$Revision: 4749 $`

**References**
```
CVE: CVE-2014-3566
BID:70574
Other:
  URL:https://www.openssl.org/~bodo/ssl-poodle.pdf
   URL:https://www.imperialviolet.org/2014/10/14/poodle.html
   URL:https://www.dfranke.us/posts/2014-10-14-how-poodle-happened.html
   URL:http://googleonlinesecurity.blogspot.in/2014/10/this-poodle-bites-exploiting-ssl-30.html
```

[ return to 127.0.0.20 ]

### 2.13.6 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

... continues on next page ...

... continued from previous page ...

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.20 ]

## 2.14 127.0.0.31

Host scan start Tue Feb 21 15:24:58 2017 UTC
Host scan end Tue Feb 21 15:53:55 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 445/tcp | High |
| 135/tcp | Medium |
| 3389/tcp | Medium |

... (continues) ...

... (continued) ...

| Service (Port) | Threat Level |
|---|---|
| general/tcp | Low |

### 2.14.1 High 445/tcp

High (CVSS: 0.0)
NVT: SMBv1 enabled (Remote Check)

**Summary**
The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.
This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices.

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

**Log Method**
Details: `SMBv1 enabled (Remote Check)`
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: `$Revision: 5222 $`

**References**
```
Other:
  URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best
↪-Practices
   URL:https://support.microsoft.com/en-us/kb/2696547
   URL:https://support.microsoft.com/en-us/kb/204279
   URL:https://technet.microsoft.com/en-us/library/security/MS17-010
```

[ return to 127.0.0.31 ]

### 2.14.2 Medium 135/tcp

Medium (CVSS: 5.0)
NVT: DCE/RPC and MSRPC Services Enumeration Reporting

**Summary**
Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker may use this fact to gain more knowledge about the remote host.

**Solution**
**Solution type:** Mitigation
Filter incoming traffic to this ports.

**Vulnerability Detection Method**
Details: `DCE/RPC and MSRPC Services Enumeration Reporting`
OID:1.3.6.1.4.1.25623.1.0.10736
Version used: `$Revision: 4998 $`

[ return to 127.0.0.31 ]

### 2.14.3 Medium 3389/tcp

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:

... continues on next page ...

... continued from previous page ...

- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Weak Cipher Suites`
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: `$Revision: 4863 $`

**References**
`CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000`
`Other:`
`URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html`
`URL:https://bettercrypto.org/`
`URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/`

Medium (CVSS: 4.0)
NVT: SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerability

**Summary**
The SSL/TLS service uses Diffie-Hellman groups with insufficient strength (key size < 2048).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to decrypt the SSL/TLS communication offline.

**Solution**
**Solution type:** Workaround
Deploy (Ephemeral) Elliptic-Curve Diffie-Hellman (ECDHE) or use a 2048-bit or stronger Diffie-Hellman group. (see https://weakdh.org/sysadmin.html).
For Apache Web Servers: Beginning with version 2.4.7, mod_ssl will use DH parameters which include primes with lengths of more than 1024 bits.

**Vulnerability Insight**
The Diffie-Hellman group are some big numbers that are used as base for the DH computations. They can be, and often are, fixed. The security of the final secret depends on the size of these parameters. It was found that 512 and 768 bits to be weak, 1024 bits to be breakable by really powerful attackers like governments.

**Vulnerability Detection Method**

... continues on next page ...

... continued from previous page ...

Checks the DHE temporary public key size.
Details: SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerabili.
↪..
OID:1.3.6.1.4.1.25623.1.0.106223
Version used: $Revision: 4739 $

**References**
Other:
URL:https://weakdh.org/
URL:https://weakdh.org/sysadmin.html

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

... continues on next page ...

... continued from previous page ...

OID:1.3.6.1.4.1.25623.1.0.105880
Version used: $Revision: 4781 $

**References**
Other:
URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with-sha-1-based-signature-algorithms/

[ return to 127.0.0.31 ]

### 2.14.4 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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

... continues on next page ...

. . . continued from previous page . . .

Version used: $Revision: 5309 $

**References**
Other:
URL:http://www.ietf.org/rfc/rfc1323.txt

[ return to 127.0.0.31 ]

## 2.15 127.0.0.34

Host scan start Tue Feb 21 16:16:42 2017 UTC
Host scan end Tue Feb 21 17:10:34 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 445/tcp | High |
| 636/tcp | Medium |
| 389/tcp | Medium |
| 3389/tcp | Medium |
| 135/tcp | Medium |
| general/tcp | Low |

### 2.15.1 High 445/tcp

High (CVSS: 0.0)
NVT: SMBv1 enabled (Remote Check)

**Summary**
The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.
This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices.

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

. . . continues on next page . . .

... continued from previous page ...

**Log Method**
Details: SMBv1 enabled (Remote Check)
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: $Revision: 5222 $

**References**
Other:
URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best-Practices
URL:https://support.microsoft.com/en-us/kb/2696547
URL:https://support.microsoft.com/en-us/kb/204279
URL:https://technet.microsoft.com/en-us/library/security/MS17-010

[ return to 127.0.0.34 ]

### 2.15.2 Medium 636/tcp

Medium (CVSS: 5.0)
NVT: Use LDAP search request to retrieve information from NT Directory Services

**Summary**
It is possible to disclose LDAP information.
Description :
The directory base of the remote server is set to NULL. This allows information to be enumerated without any prior knowledge of the directory structure.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
If pre-Windows 2000 compatibility is not required, remove pre-Windows 2000 compatibility as follows :
- start cmd.exe
- execute the command : net localgroup 'Pre-Windows 2000 Compatible Access' everyone /delete
- restart the remote host

**Vulnerability Detection Method**
Details: Use LDAP search request to retrieve information from NT Directory Services
OID:1.3.6.1.4.1.25623.1.0.12105
Version used: $Revision: 5190 $

Medium (CVSS: 5.0)
NVT: SSL/TLS: Certificate Expired

... continues on next page ...

... continued from previous page ...

**Summary**
The remote server's SSL/TLS certificate has already expired.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Replace the SSL/TLS certificate by a new one.

**Vulnerability Insight**
This script checks expiry dates of certificates associated with SSL/TLS-enabled services on the target and reports whether any have already expired.

**Vulnerability Detection Method**
Details: `SSL/TLS: Certificate Expired`
OID:1.3.6.1.4.1.25623.1.0.103955
Version used: `$Revision: 4765 $`

Medium (CVSS: 4.3)
NVT: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection

**Summary**
It was possible to detect the usage of the deprecated SSLv2 and/or SSLv3 protocol on this system.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to use the known cryptographic flaws to eavesdrop the connection between clients and the service to get access to sensitive data transferred within the secured connection.

**Solution**
**Solution type:** Mitigation
It is recommended to disable the deprecated SSLv2 and/or SSLv3 protocols in favor of the TLSv1+ protocols. Please see the references for more information.

**Affected Software/OS**
All services providing an encrypted communication using the SSLv2 and/or SSLv3 protocols.

**Vulnerability Insight**
The SSLv2 and SSLv3 protocols containing known cryptographic flaws like:
- Padding Oracle On Downgraded Legacy Encryption (POODLE, CVE-2014-3566)
- Decrypting RSA with Obsolete and Weakened eNcryption (DROWN, CVE-2016-0800)

... continues on next page ...

... continued from previous page ...

**Vulnerability Detection Method**
Check the used protocols of the services provided by this system.
Details: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection
OID:1.3.6.1.4.1.25623.1.0.111012
Version used: $Revision: 4686 $

**References**
CVE: CVE-2016-0800, CVE-2014-3566
Other:
URL:https://www.enisa.europa.eu/activities/identity-and-trust/library/deliverables/algorithms-key-sizes-and-parameters-report
URL:https://bettercrypto.org/
URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
URL:https://drownattack.com/
URL:https://www.imperialviolet.org/2014/10/14/poodle.html

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: SSL/TLS: Report Weak Cipher Suites
OID:1.3.6.1.4.1.25623.1.0.103440

... continues on next page ...

... continued from previous page ...

Version used: $Revision: 4863 $

**References**

CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000
Other:
URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html
URL:https://bettercrypto.org/
URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**

The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**

Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**

**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**

The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**

Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm
OID:1.3.6.1.4.1.25623.1.0.105880

... continues on next page ...

... continued from previous page ...

Version used: $Revision: 4781 $

**References**

Other:
URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with-sha-1-based-signature-algorithms/

[ return to 127.0.0.34 ]

### 2.15.3 Medium 389/tcp

Medium (CVSS: 5.0)
NVT: Use LDAP search request to retrieve information from NT Directory Services

**Summary**

It is possible to disclose LDAP information.
Description :
The directory base of the remote server is set to NULL. This allows information to be enumerated without any prior knowledge of the directory structure.

**Vulnerability Detection Result**

Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**

**Solution type:** Workaround
If pre-Windows 2000 compatibility is not required, remove pre-Windows 2000 compatibility as follows :
- start cmd.exe
- execute the command : net localgroup 'Pre-Windows 2000 Compatible Access' everyone /delete
- restart the remote host

**Vulnerability Detection Method**

Details: Use LDAP search request to retrieve information from NT Directory Services
OID:1.3.6.1.4.1.25623.1.0.12105
Version used: $Revision: 5190 $

[ return to 127.0.0.34 ]

### 2.15.4 Medium 3389/tcp

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**

... continues on next page ...

... continued from previous page ...

This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Weak Cipher Suites`
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: `$Revision: 4863 $`

**References**
`CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000`
`Other:`
`URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html`
`URL:https://bettercrypto.org/`
`URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/`

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

... continues on next page ...

... continued from previous page ...

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: `SSL/TLS: Certificate Signed Using A Weak Signature Algorithm`
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: `$Revision: 4781 $`

**References**
```
Other:
  URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with
↪-sha-1-based-signature-algorithms/
```

[ return to 127.0.0.34 ]

### 2.15.5 Medium 135/tcp

Medium (CVSS: 5.0)
NVT: DCE/RPC and MSRPC Services Enumeration Reporting

**Summary**
Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries.

... continues on next page ...

... continued from previous page ...

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker may use this fact to gain more knowledge about the remote host.

**Solution**
**Solution type:** Mitigation
Filter incoming traffic to this ports.

**Vulnerability Detection Method**
Details: `DCE/RPC and MSRPC Services Enumeration Reporting`
OID:1.3.6.1.4.1.25623.1.0.10736
Version used: `$Revision: 4998 $`

[ return to 127.0.0.34 ]

### 2.15.6 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

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

... continues on next page ...

. . . continued from previous page . . .

**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: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.34 ]

## 2.16 127.0.0.25

| | |
|---|---|
| Host scan start | Tue Feb 21 15:24:50 2017 UTC |
| Host scan end | Tue Feb 21 16:27:44 2017 UTC |

| Service (Port) | Threat Level |
|---|---|
| 445/tcp | High |
| 135/tcp | Medium |
| 3389/tcp | Medium |
| general/tcp | Low |

### 2.16.1 High 445/tcp

High (CVSS: 0.0)
NVT: SMBv1 enabled (Remote Check)

**Summary**
The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.
This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround

. . . continues on next page . . .

... continued from previous page ...

Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices.

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

**Log Method**
Details: `SMBv1 enabled (Remote Check)`
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: `$Revision: 5222 $`

**References**
```
Other:
  URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best
↪-Practices
   URL:https://support.microsoft.com/en-us/kb/2696547
   URL:https://support.microsoft.com/en-us/kb/204279
   URL:https://technet.microsoft.com/en-us/library/security/MS17-010
```

High (CVSS: 10.0)
NVT: Microsoft Windows SMB Server NTLM Multiple Vulnerabilities (971468)

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS10-012.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow remote attackers to execute arbitrary code or cause a denial of service or bypass the authentication mechanism via brute force technique. Impact Level: System/Application

**Solution**
**Solution type:** VendorFix
Run Windows Update and update the listed hotfixes or download and update mentioned hotfixes in the advisory from the below link, http://www.microsoft.com/technet/security/bulletin/ms10-012.mspx

**Affected Software/OS**
Microsoft Windows 7 Microsoft Windows 2000 Service Pack and prior Microsoft Windows XP Service Pack 3 and prior Microsoft Windows Vista Service Pack 2 and prior Microsoft Windows Server 2003 Service Pack 2 and prior Microsoft Windows Server 2008 Service Pack 2 and prior

**Vulnerability Insight**

... continues on next page ...

... continued from previous page ...

- An input validation error exists while processing SMB requests and can be exploited to cause a buffer overflow via a specially crafted SMB packet. - An error exists in the SMB implementation while parsing SMB packets during the Negotiate phase causing memory corruption via a specially crafted SMB packet. - NULL pointer dereference error exists in SMB while verifying the 'share' and 'servername' fields in SMB packets causing denial of service. - A lack of cryptographic entropy when the SMB server generates challenges during SMB NTLM authentication and can be exploited to bypass the authentication mechanism.

**Vulnerability Detection Method**
Details: `Microsoft Windows SMB Server NTLM Multiple Vulnerabilities (971468)`
OID:1.3.6.1.4.1.25623.1.0.902269
Version used: `$Revision: 5136 $`

**References**
`CVE: CVE-2010-0020, CVE-2010-0021, CVE-2010-0022, CVE-2010-0231`
`Other:`
`URL:http://secunia.com/advisories/38510/`
`URL:http://support.microsoft.com/kb/971468`
`URL:http://www.vupen.com/english/advisories/2010/0345`
`URL:http://www.microsoft.com/technet/security/bulletin/ms10-012.mspx`

High (CVSS: 10.0)
NVT: Microsoft Windows SMB2 Negotiation Protocol Remote Code Execution Vulnerability

**Summary**
This host is missing a critical security update according to Microsoft Bulletin MS09-050.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker can exploit this issue to execute code with SYSTEM-level privileges failed exploit attempts will likely cause denial-of-service conditions.
Impact Level: System

**Solution**
**Solution type:** VendorFix

**Affected Software/OS**
- Windows 7 RC
- Windows Vista and
- Windows 2008 Server

**Vulnerability Insight**
Multiple vulnerabilities exists,
- A denial of service vulnerability exists in the way that Microsoft Server Message Block (SMB) Protocol software handles specially crafted SMB version 2 (SMBv2) packets.

... continues on next page ...

... continued from previous page ...

- Unauthenticated remote code execution vulnerability exists in the way that Microsoft Server Message Block (SMB) Protocol software handles specially crafted SMB packets.

**Vulnerability Detection Method**
Details: `Microsoft Windows SMB2 Negotiation Protocol Remote Code Execution Vulnerability`
OID:1.3.6.1.4.1.25623.1.0.900965
Version used: `$Revision: 5074 $`

**References**
`CVE: CVE-2009-2526, CVE-2009-2532, CVE-2009-3103`
`BID:36299`
`Other:`
`URL:http://www.microsoft.com/technet/security/bulletin/MS09-050.mspx`

**Note**

```
This is a sample note on this scan result which I would like to see for any othe
↪r occurance of
this vulnerability, regardless of the task or host.
```

Last modified: Thu Mar 23 16:52:39 2017 UTC

[ return to 127.0.0.25 ]

### 2.16.2 Medium 135/tcp

Medium (CVSS: 5.0)
NVT: DCE/RPC and MSRPC Services Enumeration Reporting

**Summary**
Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker may use this fact to gain more knowledge about the remote host.

**Solution**
**Solution type:** Mitigation
Filter incoming traffic to this ports.

**Vulnerability Detection Method**

... continues on next page ...

... continued from previous page ...

Details: DCE/RPC and MSRPC Services Enumeration Reporting
OID:1.3.6.1.4.1.25623.1.0.10736
Version used: $Revision: 4998 $

[ return to 127.0.0.25 ]

### 2.16.3 Medium 3389/tcp

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: SSL/TLS: Report Weak Cipher Suites
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: $Revision: 4863 $

**References**
CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000
Other:
URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html
URL:https://bettercrypto.org/

... continues on next page ...

... continued from previous page ...

URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: $Revision: 4781 $

**References**
Other:
URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with-sha-1-based-signature-algorithms/

[ return to 127.0.0.25 ]

### 2.16.4 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.25 ]

## 2.17 127.0.0.36

| | |
|---|---|
| Host scan start | Tue Feb 21 15:24:51 2017 UTC |
| Host scan end | Tue Feb 21 16:27:23 2017 UTC |

| Service (Port) | Threat Level |
|---|---|
| 445/tcp | High |
| 135/tcp | Medium |
| general/tcp | Low |

### 2.17.1 High 445/tcp

High (CVSS: 0.0)
NVT: SMBv1 enabled (Remote Check)

**Summary**
The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.
This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices.

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

**Log Method**
Details: `SMBv1 enabled (Remote Check)`
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: `$Revision: 5222 $`

**References**
```
Other:
  URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best
↪-Practices
   URL:https://support.microsoft.com/en-us/kb/2696547
   URL:https://support.microsoft.com/en-us/kb/204279
   URL:https://technet.microsoft.com/en-us/library/security/MS17-010
```

[ return to 127.0.0.36 ]

### 2.17.2 Medium 135/tcp

| Medium (CVSS: 5.0)<br>NVT: DCE/RPC and MSRPC Services Enumeration Reporting |
|---|
| **Summary**<br>Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries. |
| **Vulnerability Detection Result**<br>Vulnerability was detected according to the Vulnerability Detection Method. |
| **Impact**<br>An attacker may use this fact to gain more knowledge about the remote host. |
| **Solution**<br>**Solution type:** Mitigation<br>Filter incoming traffic to this ports. |
| **Vulnerability Detection Method**<br>Details: `DCE/RPC and MSRPC Services Enumeration Reporting`<br>OID:1.3.6.1.4.1.25623.1.0.10736<br>Version used: `$Revision: 4998 $` |

[ return to 127.0.0.36 ]

### 2.17.3 Low general/tcp

| Low (CVSS: 2.6)<br>NVT: TCP timestamps |
|---|
| **Summary**<br>The remote host implements TCP timestamps and therefore allows to compute the uptime. |
| **Vulnerability Detection Result**<br>Vulnerability was detected according to the Vulnerability Detection Method. |
| **Impact**<br>A side effect of this feature is that the uptime of the remote host can sometimes be computed. |
| **Solution**<br>**Solution type:** Mitigation<br>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.<br>To disable TCP timestamps on Windows execute 'netsh int tcp set global timestamps=disabled' |
| . . . continues on next page . . . |

... continued from previous page ...

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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.36 ]

## 2.18 127.0.0.47

Host scan start Tue Feb 21 15:43:02 2017 UTC
Host scan end Tue Feb 21 17:34:06 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 445/tcp | High |
| general/tcp | Low |

### 2.18.1 High 445/tcp

High (CVSS: 0.0)
NVT: SMBv1 enabled (Remote Check)

**Summary**
The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.
This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810).

... continues on next page ...

... continued from previous page ...

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices.

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

**Log Method**
Details: `SMBv1 enabled (Remote Check)`
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: `$Revision: 5222 $`

**References**
```
Other:
  URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best
↪-Practices
   URL:https://support.microsoft.com/en-us/kb/2696547
   URL:https://support.microsoft.com/en-us/kb/204279
   URL:https://technet.microsoft.com/en-us/library/security/MS17-010
```

[ return to 127.0.0.47 ]

### 2.18.2 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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'

... continues on next page ...

... continued from previous page ...

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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $`

**References**
```
Other:
  URL:http://www.ietf.org/rfc/rfc1323.txt
```

[ return to 127.0.0.47 ]

## 2.19 127.0.0.8

Host scan start Tue Feb 21 16:13:58 2017 UTC
Host scan end Tue Feb 21 16:57:23 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 445/tcp | High |
| 636/tcp | High |
| 135/tcp | Medium |
| 3268/tcp | Medium |
| 443/tcp | Medium |
| 3389/tcp | Medium |
| 389/tcp | Medium |
| 3269/tcp | Medium |
| 636/tcp | Medium |
| general/tcp | Low |

### 2.19.1 High 445/tcp

High (CVSS: 0.0)
NVT: SMBv1 enabled (Remote Check)

**Summary**
The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.
This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices.

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

**Log Method**
Details: `SMBv1 enabled (Remote Check)`
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: `$Revision: 5222 $`

**References**
```
Other:
  URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best
↪-Practices
   URL:https://support.microsoft.com/en-us/kb/2696547
   URL:https://support.microsoft.com/en-us/kb/204279
   URL:https://technet.microsoft.com/en-us/library/security/MS17-010
```

[ return to 127.0.0.8 ]

### 2.19.2 High 636/tcp

High (Overridden from Medium)
NVT: Use LDAP search request to retrieve information from NT Directory Services

**Summary**
It is possible to disclose LDAP information.
Description :
The directory base of the remote server is set to NULL. This allows information to be enumerated without any prior knowledge of the directory structure.

. . . continues on next page . . .

... continued from previous page ...

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
If pre-Windows 2000 compatibility is not required, remove pre-Windows 2000 compatibility as follows :
- start cmd.exe
- execute the command : net localgroup 'Pre-Windows 2000 Compatible Access' everyone /delete
- restart the remote host

**Vulnerability Detection Method**
Details: `Use LDAP search request to retrieve information from NT Directory Services`
OID:1.3.6.1.4.1.25623.1.0.12105
Version used: `$Revision: 5190 $`

[ return to 127.0.0.8 ]

### 2.19.3 Medium 135/tcp

Medium (CVSS: 5.0)
NVT: DCE/RPC and MSRPC Services Enumeration Reporting

**Summary**
Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker may use this fact to gain more knowledge about the remote host.

**Solution**
**Solution type:** Mitigation
Filter incoming traffic to this ports.

**Vulnerability Detection Method**
Details: `DCE/RPC and MSRPC Services Enumeration Reporting`
OID:1.3.6.1.4.1.25623.1.0.10736
Version used: `$Revision: 4998 $`

[ return to 127.0.0.8 ]

### 2.19.4 Medium 3268/tcp

Medium (CVSS: 5.0)
NVT: Use LDAP search request to retrieve information from NT Directory Services

**Summary**
It is possible to disclose LDAP information.
Description :
The directory base of the remote server is set to NULL. This allows information to be enumerated without any prior knowledge of the directory structure.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
If pre-Windows 2000 compatibility is not required, remove pre-Windows 2000 compatibility as follows :
- start cmd.exe
- execute the command : net localgroup 'Pre-Windows 2000 Compatible Access' everyone /delete
- restart the remote host

**Vulnerability Detection Method**
Details: `Use LDAP search request to retrieve information from NT Directory Services`
OID:1.3.6.1.4.1.25623.1.0.12105
Version used: `$Revision: 5190 $`

[ return to 127.0.0.8 ]

### 2.19.5 Medium 443/tcp

Medium (CVSS: 5.0)
NVT: SSL/TLS: Report Vulnerable Cipher Suites for HTTPS

**Summary**
This routine reports all SSL/TLS cipher suites accepted by a service where attack vectors exists only on HTTPS services.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed cipher suites anymore.
Please see the references for more resources supporting you with this task.

. . . continues on next page . . .

... continued from previous page ...

**Affected Software/OS**
Services accepting vulnerable SSL/TLS cipher suites via HTTPS.

**Vulnerability Insight**
These rules are applied for the evaluation of the vulnerable cipher suites:
- 64-bit block cipher 3DES vulnerable to the SWEET32 attack (CVE-2016-2183).

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Vulnerable Cipher Suites for HTTPS`
OID:1.3.6.1.4.1.25623.1.0.108031
Version used: `$Revision: 5232 $`

**References**
`CVE: CVE-2016-2183, CVE-2016-6329`
`Other:`
`URL:https://bettercrypto.org/`
`URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/`
`URL:https://sweet32.info/`

Medium (CVSS: 4.3)
NVT: SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability (POODLE)

**Summary**
This host is prone to an information disclosure vulnerability.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow a man-in-the-middle attackers gain access to the plain text data stream.

**Solution**
**Solution type:** Mitigation
Possible Mitigations are:
- Disable SSLv3
- Disable cipher suites supporting CBC cipher modes
- Enable TLS_FALLBACK_SCSV if the service is providing TLSv1.0+

**Vulnerability Insight**
The flaw is due to the block cipher padding not being deterministic and not covered by the Message Authentication Code

**Vulnerability Detection Method**

... continues on next page ...

... continued from previous page ...

Evaluate previous collected information about this service.
Details: SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability .
↪..
OID:1.3.6.1.4.1.25623.1.0.802087
Version used: $Revision: 4749 $

**References**
CVE: CVE-2014-3566
BID:70574
Other:
URL:https://www.openssl.org/~bodo/ssl-poodle.pdf
URL:https://www.imperialviolet.org/2014/10/14/poodle.html
URL:https://www.dfranke.us/posts/2014-10-14-how-poodle-happened.html
URL:http://googleonlinesecurity.blogspot.in/2014/10/this-poodle-bites-exploit
↪ing-ssl-30.html

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: SSL/TLS: Report Weak Cipher Suites
OID:1.3.6.1.4.1.25623.1.0.103440

... continues on next page ...

... continued from previous page ...

Version used: $Revision: 4863 $

**References**
CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000
Other:
URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html
URL:https://bettercrypto.org/
URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/

Medium (CVSS: 4.3)
NVT: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection

**Summary**
It was possible to detect the usage of the deprecated SSLv2 and/or SSLv3 protocol on this system.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to use the known cryptographic flaws to eavesdrop the connection between clients and the service to get access to sensitive data transferred within the secured connection.

**Solution**
**Solution type:** Mitigation
It is recommended to disable the deprecated SSLv2 and/or SSLv3 protocols in favor of the TLSv1+ protocols. Please see the references for more information.

**Affected Software/OS**
All services providing an encrypted communication using the SSLv2 and/or SSLv3 protocols.

**Vulnerability Insight**
The SSLv2 and SSLv3 protocols containing known cryptographic flaws like:
- Padding Oracle On Downgraded Legacy Encryption (POODLE, CVE-2014-3566)
- Decrypting RSA with Obsolete and Weakened eNcryption (DROWN, CVE-2016-0800)

**Vulnerability Detection Method**
Check the used protocols of the services provided by this system.
Details: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection
OID:1.3.6.1.4.1.25623.1.0.111012
Version used: $Revision: 4686 $

**References**
CVE: CVE-2016-0800, CVE-2014-3566

... continues on next page ...

... continued from previous page ...

```
Other:
  URL:https://www.enisa.europa.eu/activities/identity-and-trust/library/delivera
↪bles/algorithms-key-sizes-and-parameters-report
   URL:https://bettercrypto.org/
   URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
   URL:https://drownattack.com/
   URL:https://www.imperialviolet.org/2014/10/14/poodle.html
```

Medium (CVSS: 4.0)
NVT: SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerability

**Summary**
The SSL/TLS service uses Diffie-Hellman groups with insufficient strength (key size < 2048).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to decrypt the SSL/TLS communication offline.

**Solution**
**Solution type:** Workaround
Deploy (Ephemeral) Elliptic-Curve Diffie-Hellman (ECDHE) or use a 2048-bit or stronger Diffie-Hellman group. (see https://weakdh.org/sysadmin.html).
For Apache Web Servers: Beginning with version 2.4.7, mod_ssl will use DH parameters which include primes with lengths of more than 1024 bits.

**Vulnerability Insight**
The Diffie-Hellman group are some big numbers that are used as base for the DH computations. They can be, and often are, fixed. The security of the final secret depends on the size of these parameters. It was found that 512 and 768 bits to be weak, 1024 bits to be breakable by really powerful attackers like governments.

**Vulnerability Detection Method**
Checks the DHE temporary public key size.
Details: `SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerabili.
↪..`
OID:1.3.6.1.4.1.25623.1.0.106223
Version used: `$Revision: 4739 $`

**References**

```
Other:
  URL:https://weakdh.org/
   URL:https://weakdh.org/sysadmin.html
```

[ return to 127.0.0.8 ]

### 2.19.6 Medium 3389/tcp

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Weak Cipher Suites`
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: `$Revision: 4863 $`

**References**
```
CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000
Other:
  URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-
↪1465_update_6.html
   URL:https://bettercrypto.org/
   URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
```

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**

. . . continues on next page . . .

... continued from previous page ...

The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: `SSL/TLS: Certificate Signed Using A Weak Signature Algorithm`
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: `$Revision: 4781 $`

**References**
`Other:`
`URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with-sha-1-based-signature-algorithms/`

Medium (CVSS: 4.0)
NVT: SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerability

**Summary**
The SSL/TLS service uses Diffie-Hellman groups with insufficient strength (key size < 2048).

**Vulnerability Detection Result**

... continues on next page ...

... continued from previous page ...

Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to decrypt the SSL/TLS communication offline.

**Solution**
**Solution type:** Workaround
Deploy (Ephemeral) Elliptic-Curve Diffie-Hellman (ECDHE) or use a 2048-bit or stronger Diffie-Hellman group. (see https://weakdh.org/sysadmin.html).
For Apache Web Servers: Beginning with version 2.4.7, mod_ssl will use DH parameters which include primes with lengths of more than 1024 bits.

**Vulnerability Insight**
The Diffie-Hellman group are some big numbers that are used as base for the DH computations. They can be, and often are, fixed. The security of the final secret depends on the size of these parameters. It was found that 512 and 768 bits to be weak, 1024 bits to be breakable by really powerful attackers like governments.

**Vulnerability Detection Method**
Checks the DHE temporary public key size.
Details: `SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerabili.`
`↪..`
OID:1.3.6.1.4.1.25623.1.0.106223
Version used: `$Revision: 4739 $`

**References**
```
Other:
  URL:https://weakdh.org/
   URL:https://weakdh.org/sysadmin.html
```

[ return to 127.0.0.8 ]

### 2.19.7 Medium 389/tcp

Medium (CVSS: 5.0)
NVT: Use LDAP search request to retrieve information from NT Directory Services

**Summary**
It is possible to disclose LDAP information.
Description :
The directory base of the remote server is set to NULL. This allows information to be enumerated without any prior knowledge of the directory structure.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

... continues on next page ...

... continued from previous page ...

**Solution**
**Solution type:** Workaround
If pre-Windows 2000 compatibility is not required, remove pre-Windows 2000 compatibility as follows :
- start cmd.exe
- execute the command : net localgroup 'Pre-Windows 2000 Compatible Access' everyone /delete
- restart the remote host

**Vulnerability Detection Method**
Details: `Use LDAP search request to retrieve information from NT Directory Services`
OID:1.3.6.1.4.1.25623.1.0.12105
Version used: `$Revision: 5190 $`

[ return to 127.0.0.8 ]

### 2.19.8 Medium 3269/tcp

Medium (CVSS: 5.0)
NVT: Use LDAP search request to retrieve information from NT Directory Services

**Summary**
It is possible to disclose LDAP information.
Description :
The directory base of the remote server is set to NULL. This allows information to be enumerated without any prior knowledge of the directory structure.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
If pre-Windows 2000 compatibility is not required, remove pre-Windows 2000 compatibility as follows :
- start cmd.exe
- execute the command : net localgroup 'Pre-Windows 2000 Compatible Access' everyone /delete
- restart the remote host

**Vulnerability Detection Method**
Details: `Use LDAP search request to retrieve information from NT Directory Services`
OID:1.3.6.1.4.1.25623.1.0.12105
Version used: `$Revision: 5190 $`

Medium (CVSS: 4.3)
NVT: SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability (POODLE)

**Summary**
This host is prone to an information disclosure vulnerability.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow a man-in-the-middle attackers gain access to the plain text data stream.

**Solution**
**Solution type:** Mitigation
Possible Mitigations are:
- Disable SSLv3
- Disable cipher suites supporting CBC cipher modes
- Enable TLS_FALLBACK_SCSV if the service is providing TLSv1.0+

**Vulnerability Insight**
The flaw is due to the block cipher padding not being deterministic and not covered by the Message Authentication Code

**Vulnerability Detection Method**
Evaluate previous collected information about this service.
Details: `SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability ..`
OID:1.3.6.1.4.1.25623.1.0.802087
Version used: `$Revision: 4749 $`

**References**
```
CVE: CVE-2014-3566
BID:70574
Other:
  URL:https://www.openssl.org/~bodo/ssl-poodle.pdf
   URL:https://www.imperialviolet.org/2014/10/14/poodle.html
   URL:https://www.dfranke.us/posts/2014-10-14-how-poodle-happened.html
   URL:http://googleonlinesecurity.blogspot.in/2014/10/this-poodle-bites-exploiting-ssl-30.html
```

Medium (CVSS: 4.3)
NVT: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection

**Summary**

. . . continues on next page . . .

... continued from previous page ...

It was possible to detect the usage of the deprecated SSLv2 and/or SSLv3 protocol on this system.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to use the known cryptographic flaws to eavesdrop the connection between clients and the service to get access to sensitive data transferred within the secured connection.

**Solution**
**Solution type:** Mitigation
It is recommended to disable the deprecated SSLv2 and/or SSLv3 protocols in favor of the TLSv1+ protocols. Please see the references for more information.

**Affected Software/OS**
All services providing an encrypted communication using the SSLv2 and/or SSLv3 protocols.

**Vulnerability Insight**
The SSLv2 and SSLv3 protocols containing known cryptographic flaws like:
- Padding Oracle On Downgraded Legacy Encryption (POODLE, CVE-2014-3566)
- Decrypting RSA with Obsolete and Weakened eNcryption (DROWN, CVE-2016-0800)

**Vulnerability Detection Method**
Check the used protocols of the services provided by this system.
Details: `SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection`
OID:1.3.6.1.4.1.25623.1.0.111012
Version used: `$Revision: 4686 $`

**References**
```
CVE: CVE-2016-0800, CVE-2014-3566
Other:
  URL:https://www.enisa.europa.eu/activities/identity-and-trust/library/delivera
↪bles/algorithms-key-sizes-and-parameters-report
   URL:https://bettercrypto.org/
   URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
   URL:https://drownattack.com/
   URL:https://www.imperialviolet.org/2014/10/14/poodle.html
```

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.

... continues on next page ...

... continued from previous page ...

NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Weak Cipher Suites`
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: `$Revision: 4863 $`

**References**
`CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000`
`Other:`
`URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html`
`URL:https://bettercrypto.org/`
`URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/`

Medium (CVSS: 4.0)
NVT: SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerability

**Summary**
The SSL/TLS service uses Diffie-Hellman groups with insufficient strength (key size < 2048).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to decrypt the SSL/TLS communication offline.

... continues on next page ...

... continued from previous page ...

**Solution**
**Solution type:** Workaround
Deploy (Ephemeral) Elliptic-Curve Diffie-Hellman (ECDHE) or use a 2048-bit or stronger Diffie-Hellman group. (see https://weakdh.org/sysadmin.html).
For Apache Web Servers: Beginning with version 2.4.7, mod_ssl will use DH parameters which include primes with lengths of more than 1024 bits.

**Vulnerability Insight**
The Diffie-Hellman group are some big numbers that are used as base for the DH computations. They can be, and often are, fixed. The security of the final secret depends on the size of these parameters. It was found that 512 and 768 bits to be weak, 1024 bits to be breakable by really powerful attackers like governments.

**Vulnerability Detection Method**
Checks the DHE temporary public key size.
Details: `SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerabili.`
`↪..`
OID:1.3.6.1.4.1.25623.1.0.106223
Version used: `$Revision: 4739 $`

**References**
```
Other:
  URL:https://weakdh.org/
   URL:https://weakdh.org/sysadmin.html
```

[ return to 127.0.0.8 ]

### 2.19.9 Medium 636/tcp

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.

... continues on next page ...

... continued from previous page ...

Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Weak Cipher Suites`
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: `$Revision: 4863 $`

**References**
```
CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000
Other:
  URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-
↪1465_update_6.html
   URL:https://bettercrypto.org/
   URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
```

## Medium (CVSS: 4.3)
## NVT: SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability (POODLE)

**Summary**
This host is prone to an information disclosure vulnerability.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow a man-in-the-middle attackers gain access to the plain text data stream.

**Solution**
**Solution type:** Mitigation
Possible Mitigations are:
- Disable SSLv3
- Disable cipher suites supporting CBC cipher modes
- Enable TLS_FALLBACK_SCSV if the service is providing TLSv1.0+

**Vulnerability Insight**

... continues on next page ...

... continued from previous page ...

The flaw is due to the block cipher padding not being deterministic and not covered by the Message Authentication Code

**Vulnerability Detection Method**
Evaluate previous collected information about this service.
Details: SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability ...
OID:1.3.6.1.4.1.25623.1.0.802087
Version used: $Revision: 4749 $

**References**
CVE: CVE-2014-3566
BID:70574
Other:
URL:https://www.openssl.org/~bodo/ssl-poodle.pdf
URL:https://www.imperialviolet.org/2014/10/14/poodle.html
URL:https://www.dfranke.us/posts/2014-10-14-how-poodle-happened.html
URL:http://googleonlinesecurity.blogspot.in/2014/10/this-poodle-bites-exploiting-ssl-30.html

Medium (CVSS: 4.3)
NVT: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection

**Summary**
It was possible to detect the usage of the deprecated SSLv2 and/or SSLv3 protocol on this system.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to use the known cryptographic flaws to eavesdrop the connection between clients and the service to get access to sensitive data transferred within the secured connection.

**Solution**
**Solution type:** Mitigation
It is recommended to disable the deprecated SSLv2 and/or SSLv3 protocols in favor of the TLSv1+ protocols. Please see the references for more information.

**Affected Software/OS**
All services providing an encrypted communication using the SSLv2 and/or SSLv3 protocols.

**Vulnerability Insight**
The SSLv2 and SSLv3 protocols containing known cryptographic flaws like:
- Padding Oracle On Downgraded Legacy Encryption (POODLE, CVE-2014-3566)

... continues on next page ...

... continued from previous page ...

- Decrypting RSA with Obsolete and Weakened eNcryption (DROWN, CVE-2016-0800)

**Vulnerability Detection Method**
Check the used protocols of the services provided by this system.
Details: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection
OID:1.3.6.1.4.1.25623.1.0.111012
Version used: $Revision: 4686 $

**References**
CVE: CVE-2016-0800, CVE-2014-3566
Other:
URL:https://www.enisa.europa.eu/activities/identity-and-trust/library/deliverables/algorithms-key-sizes-and-parameters-report
URL:https://bettercrypto.org/
URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
URL:https://drownattack.com/
URL:https://www.imperialviolet.org/2014/10/14/poodle.html

Medium (CVSS: 4.0)
NVT: SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerability

**Summary**
The SSL/TLS service uses Diffie-Hellman groups with insufficient strength (key size < 2048).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to decrypt the SSL/TLS communication offline.

**Solution**
**Solution type:** Workaround
Deploy (Ephemeral) Elliptic-Curve Diffie-Hellman (ECDHE) or use a 2048-bit or stronger Diffie-Hellman group. (see https://weakdh.org/sysadmin.html).
For Apache Web Servers: Beginning with version 2.4.7, mod_ssl will use DH parameters which include primes with lengths of more than 1024 bits.

**Vulnerability Insight**
The Diffie-Hellman group are some big numbers that are used as base for the DH computations. They can be, and often are, fixed. The security of the final secret depends on the size of these parameters. It was found that 512 and 768 bits to be weak, 1024 bits to be breakable by really powerful attackers like governments.

**Vulnerability Detection Method**
Checks the DHE temporary public key size.
Details: SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerabili.

... continues on next page ...

... continued from previous page ...

↪..
OID:1.3.6.1.4.1.25623.1.0.106223
Version used: $Revision: 4739 $

**References**
Other:
URL:https://weakdh.org/
URL:https://weakdh.org/sysadmin.html

[ return to 127.0.0.8 ]

### 2.19.10 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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

... continues on next page ...

... continued from previous page ...

OID:1.3.6.1.4.1.25623.1.0.80091
Version used: $Revision: 5309 $

**References**
Other:
URL:http://www.ietf.org/rfc/rfc1323.txt

[ return to 127.0.0.8 ]

## 2.20 127.0.0.35

Host scan start Tue Feb 21 15:24:51 2017 UTC
Host scan end Tue Feb 21 15:35:23 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 445/tcp | High |
| general/tcp | High |

### 2.20.1 High 445/tcp

High (CVSS: 0.0)
NVT: SMBv1 enabled (Remote Check)

**Summary**
The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.
This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices.

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

**Log Method**
Details: SMBv1 enabled (Remote Check)
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: $Revision: 5222 $

... continues on next page ...

... continued from previous page ...

**References**
Other:
URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best-Practices
URL:https://support.microsoft.com/en-us/kb/2696547
URL:https://support.microsoft.com/en-us/kb/204279
URL:https://technet.microsoft.com/en-us/library/security/MS17-010

[ return to 127.0.0.35 ]

### 2.20.2 High general/tcp

High (CVSS: 10.0)
NVT: OS End Of Life Detection

**Summary**
OS End Of Life Detection
The Operating System on the remote host has reached the end of life and should not be used anymore.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation

**Vulnerability Detection Method**
Details: OS End Of Life Detection
OID:1.3.6.1.4.1.25623.1.0.103674
Version used: $Revision: 4111 $

[ return to 127.0.0.35 ]

## 2.21 127.0.0.39

Host scan start Tue Feb 21 15:35:50 2017 UTC
Host scan end Tue Feb 21 17:13:29 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 445/tcp | High |
| 135/tcp | Medium |
| 3389/tcp | Medium |
| general/tcp | Low |

### 2.21.1 High 445/tcp

| High (CVSS: 0.0)<br>NVT: SMBv1 enabled (Remote Check) |
|---|
| **Summary**<br>The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.<br>This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810). |
| **Vulnerability Detection Result**<br>Vulnerability was detected according to the Vulnerability Detection Method. |
| **Solution**<br>**Solution type:** Workaround<br>Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices. |
| **Vulnerability Insight**<br>The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group. |
| **Log Method**<br>Details: `SMBv1 enabled (Remote Check)`<br>OID:1.3.6.1.4.1.25623.1.0.140151<br>Version used: `$Revision: 5222 $` |
| **References**<br>`Other:`<br>`URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best-Practices`<br>`URL:https://support.microsoft.com/en-us/kb/2696547`<br>`URL:https://support.microsoft.com/en-us/kb/204279`<br>`URL:https://technet.microsoft.com/en-us/library/security/MS17-010` |

[ return to 127.0.0.39 ]

### 2.21.2 Medium 135/tcp

| Medium (CVSS: 5.0)<br>NVT: DCE/RPC and MSRPC Services Enumeration Reporting |
|---|
| **Summary** |
| ... continues on next page ... |

... continued from previous page ...

Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker may use this fact to gain more knowledge about the remote host.

**Solution**
**Solution type:** Mitigation
Filter incoming traffic to this ports.

**Vulnerability Detection Method**
Details: `DCE/RPC and MSRPC Services Enumeration Reporting`
OID:1.3.6.1.4.1.25623.1.0.10736
Version used: `$Revision: 4998 $`

[ return to 127.0.0.39 ]

### 2.21.3 Medium 3389/tcp

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).

... continues on next page ...

... continued from previous page ...

- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Weak Cipher Suites`
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: `$Revision: 4863 $`

**References**
`CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000`
`Other:`
`URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html`
`URL:https://bettercrypto.org/`
`URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/`

[ return to 127.0.0.39 ]

### 2.21.4 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

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

... continues on next page ...

. . . continued from previous page . . .

**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: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.39 ]

## 2.22 127.0.0.2

| | |
|---|---|
| Host scan start | Tue Feb 21 15:24:58 2017 UTC |
| Host scan end | Tue Feb 21 16:21:25 2017 UTC |

| Service (Port) | Threat Level |
|---|---|
| 445/tcp | High |
| 3389/tcp | Medium |
| 135/tcp | Medium |
| general/tcp | Low |

### 2.22.1 High 445/tcp

High (CVSS: 0.0)
NVT: SMBv1 enabled (Remote Check)

**Summary**
The remote Windows host is prone to an unspecified remote code execution vulnerability in SMBv1 protocol.
This NVT has been replaced by NVT 'Microsoft Windows SMB Server Multiple Vulnerabilities-Remote (4013389)' (OID: 1.3.6.1.4.1.25623.1.0.810810).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround

. . . continues on next page . . .

... continued from previous page ...

Disable SMB v1 and/or block all versions of SMB at the network boundary by blocking TCP port 445 with related protocols on UDP ports 137-138 and TCP port 139, for all boundary devices.

**Vulnerability Insight**
The remote Windows host is supporting SMBv1 and is therefore prone to an unspecified remote code execution vulnerability. This vulnerability is related to the 'Shadow Brokers' group.

**Log Method**
Details: `SMBv1 enabled (Remote Check)`
OID:1.3.6.1.4.1.25623.1.0.140151
Version used: `$Revision: 5222 $`

**References**
```
Other:
  URL:https://www.us-cert.gov/ncas/current-activity/2017/01/16/SMB-Security-Best
↪-Practices
   URL:https://support.microsoft.com/en-us/kb/2696547
   URL:https://support.microsoft.com/en-us/kb/204279
   URL:https://technet.microsoft.com/en-us/library/security/MS17-010
```

[ return to 127.0.0.2 ]

### 2.22.2 Medium 3389/tcp

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).

... continues on next page ...

... continued from previous page ...

- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Weak Cipher Suites`
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: `$Revision: 4863 $`

**References**
`CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000`
`Other:`
`URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html`
`URL:https://bettercrypto.org/`
`URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/`

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.

... continues on next page ...

... continued from previous page ...

NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: `SSL/TLS: Certificate Signed Using A Weak Signature Algorithm`
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: `$Revision: 4781 $`

**References**
`Other:`
`URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with-sha-1-based-signature-algorithms/`

Medium (CVSS: 4.0)
NVT: SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerability

**Summary**
The SSL/TLS service uses Diffie-Hellman groups with insufficient strength (key size < 2048).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to decrypt the SSL/TLS communication offline.

**Solution**
**Solution type:** Workaround
Deploy (Ephemeral) Elliptic-Curve Diffie-Hellman (ECDHE) or use a 2048-bit or stronger Diffie-Hellman group. (see https://weakdh.org/sysadmin.html).
For Apache Web Servers: Beginning with version 2.4.7, mod_ssl will use DH parameters which include primes with lengths of more than 1024 bits.

**Vulnerability Insight**
The Diffie-Hellman group are some big numbers that are used as base for the DH computations. They can be, and often are, fixed. The security of the final secret depends on the size of these parameters. It was found that 512 and 768 bits to be weak, 1024 bits to be breakable by really powerful attackers like governments.

**Vulnerability Detection Method**
Checks the DHE temporary public key size.
Details: `SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerabili.`

... continues on next page ...

... continued from previous page ...

↪..
OID:1.3.6.1.4.1.25623.1.0.106223
Version used: $Revision: 4739 $

**References**
Other:
URL:https://weakdh.org/
URL:https://weakdh.org/sysadmin.html

[ return to 127.0.0.2 ]

### 2.22.3 Medium 135/tcp

Medium (CVSS: 5.0)
NVT: DCE/RPC and MSRPC Services Enumeration Reporting

**Summary**
Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker may use this fact to gain more knowledge about the remote host.

**Solution**
**Solution type:** Mitigation
Filter incoming traffic to this ports.

**Vulnerability Detection Method**
Details: DCE/RPC and MSRPC Services Enumeration Reporting
OID:1.3.6.1.4.1.25623.1.0.10736
Version used: $Revision: 4998 $

[ return to 127.0.0.2 ]

### 2.22.4 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

**Summary**

... continues on next page ...

... continued from previous page ...

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $`

**References**
```
Other:
  URL:http://www.ietf.org/rfc/rfc1323.txt
```

[ return to 127.0.0.2 ]

## 2.23 127.0.0.6

Host scan start Tue Feb 21 15:58:23 2017 UTC
Host scan end Tue Feb 21 16:42:51 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 22/tcp | High |
| 9390/tcp | Medium |

... (continues) ...

... (continued) ...

| Service (Port) | Threat Level |
|---|---|
| 22/tcp | Medium |
| 443/tcp | Medium |
| general/tcp | Low |
| 22/tcp | Low |

### 2.23.1 High 22/tcp

High (CVSS: 7.5)
NVT: SSH Brute Force Logins With Default Credentials Reporting

**Summary**
It was possible to login into the remote SSH server using default credentials.
As the NVT 'SSH Brute Force Logins with default Credentials' (OID: 1.3.6.1.4.1.25623.1.0.108013) might run into a timeout the actual reporting of this vulnerability takes place in this NVT instead. The script preference 'Report timeout' allows you to configure if such an timeout is reported.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Change the password as soon as possible.

**Vulnerability Detection Method**
Try to login with a number of known default credentials via the SSH protocol.
Details: `SSH Brute Force Logins With Default Credentials Reporting`
OID:1.3.6.1.4.1.25623.1.0.103239
Version used: `$Revision: 4508 $`

[ return to 127.0.0.6 ]

### 2.23.2 Medium 9390/tcp

Medium (CVSS: 5.0)
NVT: SSL/TLS: Certificate Expired

**Summary**
The remote server's SSL/TLS certificate has already expired.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

... continues on next page ...

... continued from previous page ...

**Solution**
**Solution type:** Mitigation
Replace the SSL/TLS certificate by a new one.

**Vulnerability Insight**
This script checks expiry dates of certificates associated with SSL/TLS-enabled services on the target and reports whether any have already expired.

**Vulnerability Detection Method**
Details: `SSL/TLS: Certificate Expired`
OID:1.3.6.1.4.1.25623.1.0.103955
Version used: `$Revision: 4765 $`

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**

... continues on next page ...

... continued from previous page ...

Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: $Revision: 4781 $

**References**
Other:
URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with-sha-1-based-signature-algorithms/

[ return to 127.0.0.6 ]

### 2.23.3 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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: 4490 $

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

[ return to 127.0.0.6 ]

### 2.23.4 Medium 443/tcp

Medium (CVSS: 5.0)
NVT: SSL/TLS: Certificate Expired

**Summary**
The remote server's SSL/TLS certificate has already expired.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Replace the SSL/TLS certificate by a new one.

**Vulnerability Insight**
This script checks expiry dates of certificates associated with SSL/TLS-enabled services on the target and reports whether any have already expired.

**Vulnerability Detection Method**
Details: `SSL/TLS: Certificate Expired`
OID:1.3.6.1.4.1.25623.1.0.103955
Version used: `$Revision: 4765 $`

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)

. . . continues on next page . . .

... continued from previous page ...

- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)

Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: `SSL/TLS: Certificate Signed Using A Weak Signature Algorithm`
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: `$Revision: 4781 $`

**References**
`Other:`
`URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with-sha-1-based-signature-algorithms/`

[ return to 127.0.0.6 ]

### 2.23.5 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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.

... continues on next page ...

... continued from previous page ...

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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.6 ]

### 2.23.6 Low 22/tcp

Low (CVSS: 2.6)
NVT: SSH Weak MAC Algorithms Supported

**Summary**
The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Disable the weak MAC algorithms.

**Vulnerability Detection Method**
Details: `SSH Weak MAC Algorithms Supported`
OID:1.3.6.1.4.1.25623.1.0.105610
Version used: `$Revision: 4490 $`

[ return to 127.0.0.6 ]

## 2.24 127.0.0.3

| | |
|---|---|
| Host scan start | Tue Feb 21 15:58:36 2017 UTC |
| Host scan end | Tue Feb 21 16:57:15 2017 UTC |

| Service (Port) | Threat Level |
|---|---|
| 22/tcp | High |
| 443/tcp | Medium |
| 9390/tcp | Medium |
| general/tcp | Low |

### 2.24.1 High 22/tcp

High (CVSS: 7.5)
NVT: SSH Brute Force Logins With Default Credentials Reporting

**Summary**
It was possible to login into the remote SSH server using default credentials.
As the NVT 'SSH Brute Force Logins with default Credentials' (OID: 1.3.6.1.4.1.25623.1.0.108013) might run into a timeout the actual reporting of this vulnerability takes place in this NVT instead. The script preference 'Report timeout' allows you to configure if such an timeout is reported.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Change the password as soon as possible.

**Vulnerability Detection Method**
Try to login with a number of known default credentials via the SSH protocol.
Details: `SSH Brute Force Logins With Default Credentials Reporting`
OID:1.3.6.1.4.1.25623.1.0.103239
Version used: `$Revision: 4508 $`

[ return to 127.0.0.3 ]

### 2.24.2 Medium 443/tcp

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

. . . continues on next page . . .

... continued from previous page ...

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: `SSL/TLS: Certificate Signed Using A Weak Signature Algorithm`
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: `$Revision: 4781 $`

**References**
```
Other:
  URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with
↪-sha-1-based-signature-algorithms/
```

[ return to 127.0.0.3 ]

### 2.24.3 Medium 9390/tcp

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**

... continues on next page ...

... continued from previous page ...

The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: `SSL/TLS: Certificate Signed Using A Weak Signature Algorithm`
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: `$Revision: 4781 $`

**References**
`Other:`
`URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with-sha-1-based-signature-algorithms/`

[ return to 127.0.0.3 ]

### 2.24.4 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.3 ]

## 2.25 127.0.0.43

| | |
|---|---|
| Host scan start | Tue Feb 21 16:23:30 2017 UTC |
| Host scan end | Tue Feb 21 17:04:31 2017 UTC |

| Service (Port) | Threat Level |
|---|---|
| 22/tcp | High |

### 2.25.1 High 22/tcp

High (CVSS: 7.5)
NVT: SSH Brute Force Logins With Default Credentials Reporting

**Summary**
It was possible to login into the remote SSH server using default credentials.
As the NVT 'SSH Brute Force Logins with default Credentials' (OID: 1.3.6.1.4.1.25623.1.0.108013) might run into a timeout the actual reporting of this vulnerability takes place in this NVT instead. The script preference 'Report timeout' allows you to configure if such an timeout is reported.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Change the password as soon as possible.

**Vulnerability Detection Method**
Try to login with a number of known default credentials via the SSH protocol.
Details: `SSH Brute Force Logins With Default Credentials Reporting`
OID:1.3.6.1.4.1.25623.1.0.103239
Version used: `$Revision: 4508 $`

[ return to 127.0.0.43 ]

## 2.26 127.0.0.28

Host scan start Tue Feb 21 15:24:41 2017 UTC
Host scan end Tue Feb 21 18:02:20 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 22/tcp | High |
| general/tcp | Low |

### 2.26.1 High 22/tcp

High (CVSS: 7.5)
NVT: SSH Brute Force Logins With Default Credentials Reporting

. . . continues on next page . . .

... continued from previous page ...

**Summary**
It was possible to login into the remote SSH server using default credentials.
As the NVT 'SSH Brute Force Logins with default Credentials' (OID: 1.3.6.1.4.1.25623.1.0.108013) might run into a timeout the actual reporting of this vulnerability takes place in this NVT instead. The script preference 'Report timeout' allows you to configure if such an timeout is reported.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Change the password as soon as possible.

**Vulnerability Detection Method**
Try to login with a number of known default credentials via the SSH protocol.
Details: `SSH Brute Force Logins With Default Credentials Reporting`
OID:1.3.6.1.4.1.25623.1.0.103239
Version used: `$Revision: 4508 $`

[ return to 127.0.0.28 ]

### 2.26.2 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

... continues on next page ...

... continued from previous page ...

**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: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.28 ]

## 2.27 127.0.0.32

| | |
|---|---|
| Host scan start | Tue Feb 21 16:04:02 2017 UTC |
| Host scan end | Tue Feb 21 16:49:36 2017 UTC |

| Service (Port) | Threat Level |
|---|---|
| 22/tcp | High |
| general/tcp | Low |

### 2.27.1 High 22/tcp

High (CVSS: 7.5)
NVT: SSH Brute Force Logins With Default Credentials Reporting

**Summary**
It was possible to login into the remote SSH server using default credentials.
As the NVT 'SSH Brute Force Logins with default Credentials' (OID: 1.3.6.1.4.1.25623.1.0.108013) might run into a timeout the actual reporting of this vulnerability takes place in this NVT instead. The script preference 'Report timeout' allows you to configure if such an timeout is reported.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**

... continues on next page ...

... continued from previous page ...

**Solution type:** Mitigation
Change the password as soon as possible.

**Vulnerability Detection Method**
Try to login with a number of known default credentials via the SSH protocol.
Details: `SSH Brute Force Logins With Default Credentials Reporting`
OID:1.3.6.1.4.1.25623.1.0.103239
Version used: `$Revision: 4508 $`

[ return to 127.0.0.32 ]

### 2.27.2 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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`

... continues on next page ...

... continued from previous page ...

OID:1.3.6.1.4.1.25623.1.0.80091
Version used: `$Revision: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.32 ]

## 2.28 127.0.0.5

Host scan start Tue Feb 21 15:54:03 2017 UTC
Host scan end Tue Feb 21 16:44:32 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 901/tcp | High |
| 22/tcp | Medium |
| 22/tcp | Low |
| general/tcp | Low |

### 2.28.1 High 901/tcp

High (CVSS: 7.2)
NVT: Detect SWAT server port

**Summary**
SWAT (Samba Web Administration Tool) is running on this port.
SWAT allows Samba users to change their passwords, and offers to the sysadmin an easy-to-use GUI to configure Samba.
However, it is not recommended to let SWAT be accessed by the world, as it allows an intruder to attempt to brute force some accounts passwords.
In addition to this, the traffic between SWAT and web clients is not ciphered, so an eavesdropper can gain clear text passwords easily.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
Disable SWAT access from the outside network by making your firewall filter this port.
If you do not need SWAT, disable it by commenting the relevant /etc/inetd.conf line.

**Vulnerability Detection Method**
Details: `Detect SWAT server port`
OID:1.3.6.1.4.1.25623.1.0.10273

... continues on next page ...

... continued from previous page ...

Version used: $Revision: 3362 $

**References**
CVE: CVE-2000-0935
BID:1872

[ return to 127.0.0.5 ]

### 2.28.2 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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: 4490 $

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

[ return to 127.0.0.5 ]

### 2.28.3 Low 22/tcp

| Low (CVSS: 2.6)<br>NVT: SSH Weak MAC Algorithms Supported |
|---|
| **Summary**<br>The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms. |
| **Vulnerability Detection Result**<br>Vulnerability was detected according to the Vulnerability Detection Method. |
| **Solution**<br>**Solution type:** Mitigation<br>Disable the weak MAC algorithms. |
| **Vulnerability Detection Method**<br>Details: `SSH Weak MAC Algorithms Supported`<br>OID:1.3.6.1.4.1.25623.1.0.105610<br>Version used: `$Revision: 4490 $` |

[ return to 127.0.0.5 ]

### 2.28.4 Low general/tcp

| Low (CVSS: 2.6)<br>NVT: TCP timestamps |
|---|
| **Summary**<br>The remote host implements TCP timestamps and therefore allows to compute the uptime. |
| **Vulnerability Detection Result**<br>Vulnerability was detected according to the Vulnerability Detection Method. |
| **Impact**<br>A side effect of this feature is that the uptime of the remote host can sometimes be computed. |
| **Solution**<br>**Solution type:** Mitigation<br>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.<br>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.<br>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.<br>See also: http://www.microsoft.com/en-us/download/details.aspx?id=9152 |
| . . . continues on next page . . . |

... continued from previous page ...

**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: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.5 ]

## 2.29 127.0.0.38

| | |
|---|---|
| Host scan start | Tue Feb 21 15:24:45 2017 UTC |
| Host scan end | Tue Feb 21 15:55:50 2017 UTC |

| Service (Port) | Threat Level |
|---|---|
| 901/tcp | High |
| 22/tcp | Medium |
| general/tcp | Low |
| 22/tcp | Low |

### 2.29.1 High 901/tcp

High (CVSS: 7.2)
NVT: Detect SWAT server port

**Summary**
SWAT (Samba Web Administration Tool) is running on this port.
SWAT allows Samba users to change their passwords, and offers to the sysadmin an easy-to-use GUI to configure Samba.
However, it is not recommended to let SWAT be accessed by the world, as it allows an intruder to attempt to brute force some accounts passwords.
In addition to this, the traffic between SWAT and web clients is not ciphered, so an eavesdropper can gain clear text passwords easily.

... continues on next page ...

... continued from previous page ...

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Workaround
Disable SWAT access from the outside network by making your firewall filter this port.
If you do not need SWAT, disable it by commenting the relevant /etc/inetd.conf line.

**Vulnerability Detection Method**
Details: `Detect SWAT server port`
OID:1.3.6.1.4.1.25623.1.0.10273
Version used: `$Revision: 3362 $`

**References**
`CVE: CVE-2000-0935`
`BID:1872`

[ return to 127.0.0.38 ]

### 2.29.2 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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.

... continues on next page ...

... continued from previous page ...

Details: SSH Weak Encryption Algorithms Supported
OID:1.3.6.1.4.1.25623.1.0.105611
Version used: $Revision: 4490 $

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

[ return to 127.0.0.38 ]

### 2.29.3 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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

... continues on next page ...

... continued from previous page ...

OID:1.3.6.1.4.1.25623.1.0.80091
Version used: `$Revision: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.38 ]

### 2.29.4 Low 22/tcp

Low (CVSS: 2.6)
NVT: SSH Weak MAC Algorithms Supported

**Summary**
The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Disable the weak MAC algorithms.

**Vulnerability Detection Method**
Details: `SSH Weak MAC Algorithms Supported`
OID:1.3.6.1.4.1.25623.1.0.105610
Version used: `$Revision: 4490 $`

[ return to 127.0.0.38 ]

## 2.30 127.0.0.41

Host scan start Tue Feb 21 15:24:38 2017 UTC
Host scan end Tue Feb 21 16:02:04 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 443/tcp | Medium |
| 22/tcp | Medium |

### 2.30.1 Medium 443/tcp

Medium (CVSS: 6.8)
NVT: SSL/TLS: OpenSSL CCS Man in the Middle Security Bypass Vulnerability

**Summary**
OpenSSL is prone to security-bypass vulnerability.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successfully exploiting this issue may allow attackers to obtain sensitive information by conducting a man-in-the-middle attack. This may lead to other attacks.

**Solution**
**Solution type:** VendorFix
Updates are available.

**Affected Software/OS**
OpenSSL before 0.9.8za, 1.0.0 before 1.0.0m and 1.0.1 before 1.0.1h

**Vulnerability Insight**
OpenSSL does not properly restrict processing of ChangeCipherSpec messages, which allows man-in-the-middle attackers to trigger use of a zero-length master key in certain OpenSSL-to-OpenSSL communications, and consequently hijack sessions or obtain sensitive information, via a crafted TLS handshake, aka the 'CCS Injection' vulnerability.

**Vulnerability Detection Method**
Send two SSL ChangeCipherSpec request and check the response.
Details: `SSL/TLS: OpenSSL CCS Man in the Middle Security Bypass Vulnerability`
OID:1.3.6.1.4.1.25623.1.0.105042
Version used: `$Revision: 4679 $`

**References**
`CVE: CVE-2014-0224`
`BID:67899`
`Other:`
`URL:http://www.securityfocus.com/bid/67899`
`URL:http://openssl.org/`

Medium (CVSS: 5.0)
NVT: SSL/TLS: Report Vulnerable Cipher Suites for HTTPS

**Summary**
This routine reports all SSL/TLS cipher suites accepted by a service where attack vectors exists only on HTTPS services.

**Vulnerability Detection Result**

. . . continues on next page . . .

... continued from previous page ...

Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Affected Software/OS**
Services accepting vulnerable SSL/TLS cipher suites via HTTPS.

**Vulnerability Insight**
These rules are applied for the evaluation of the vulnerable cipher suites:
- 64-bit block cipher 3DES vulnerable to the SWEET32 attack (CVE-2016-2183).

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Vulnerable Cipher Suites for HTTPS`
OID:1.3.6.1.4.1.25623.1.0.108031
Version used: `$Revision: 5232 $`

**References**
```
CVE: CVE-2016-2183, CVE-2016-6329
Other:
  URL:https://bettercrypto.org/
   URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
   URL:https://sweet32.info/
```

Medium (CVSS: 4.3)
NVT: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection

**Summary**
It was possible to detect the usage of the deprecated SSLv2 and/or SSLv3 protocol on this system.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to use the known cryptographic flaws to eavesdrop the connection between clients and the service to get access to sensitive data transferred within the secured connection.

**Solution**
**Solution type:** Mitigation
It is recommended to disable the deprecated SSLv2 and/or SSLv3 protocols in favor of the TLSv1+ protocols. Please see the references for more information.

... continues on next page ...

... continued from previous page ...

**Affected Software/OS**
All services providing an encrypted communication using the SSLv2 and/or SSLv3 protocols.

**Vulnerability Insight**
The SSLv2 and SSLv3 protocols containing known cryptographic flaws like:
- Padding Oracle On Downgraded Legacy Encryption (POODLE, CVE-2014-3566)
- Decrypting RSA with Obsolete and Weakened eNcryption (DROWN, CVE-2016-0800)

**Vulnerability Detection Method**
Check the used protocols of the services provided by this system.
Details: `SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection`
OID:1.3.6.1.4.1.25623.1.0.111012
Version used: `$Revision: 4686 $`

**References**
```
CVE: CVE-2016-0800, CVE-2014-3566
Other:
  URL:https://www.enisa.europa.eu/activities/identity-and-trust/library/delivera
↪bles/algorithms-key-sizes-and-parameters-report
   URL:https://bettercrypto.org/
   URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
   URL:https://drownattack.com/
   URL:https://www.imperialviolet.org/2014/10/14/poodle.html
```

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)

... continues on next page ...

... continued from previous page ...

- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: `SSL/TLS: Certificate Signed Using A Weak Signature Algorithm`
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: `$Revision: 4781 $`

**References**
```
Other:
  URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with
↪-sha-1-based-signature-algorithms/
```

[ return to 127.0.0.41 ]

### 2.30.2 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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.

... continues on next page ...

. . . continued from previous page . . .

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: 4490 $`

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

[ return to 127.0.0.41 ]

## 2.31 127.0.0.27

Host scan start Tue Feb 21 16:27:15 2017 UTC
Host scan end Tue Feb 21 17:58:01 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 22/tcp | Medium |
| 3871/tcp | Medium |
| 22/tcp | Low |
| general/tcp | Low |

### 2.31.1 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Disable the weak encryption algorithms.

**Vulnerability Insight**

. . . continues on next page . . .

... continued from previous page ...

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: 4490 $`

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

[ return to 127.0.0.27 ]

### 2.31.2 Medium 3871/tcp

Medium (CVSS: 6.8)
NVT: SSL/TLS: OpenSSL CCS Man in the Middle Security Bypass Vulnerability

**Summary**
OpenSSL is prone to security-bypass vulnerability.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successfully exploiting this issue may allow attackers to obtain sensitive information by conducting a man-in-the-middle attack. This may lead to other attacks.

**Solution**
**Solution type:** VendorFix
Updates are available.

**Affected Software/OS**
OpenSSL before 0.9.8za, 1.0.0 before 1.0.0m and 1.0.1 before 1.0.1h

**Vulnerability Insight**

... continues on next page ...

... continued from previous page ...

OpenSSL does not properly restrict processing of ChangeCipherSpec messages, which allows man-in-the-middle attackers to trigger use of a zero-length master key in certain OpenSSL-to-OpenSSL communications, and consequently hijack sessions or obtain sensitive information, via a crafted TLS handshake, aka the 'CCS Injection' vulnerability.

**Vulnerability Detection Method**
Send two SSL ChangeCipherSpec request and check the response.
Details: `SSL/TLS: OpenSSL CCS Man in the Middle Security Bypass Vulnerability`
OID:1.3.6.1.4.1.25623.1.0.105042
Version used: `$Revision: 4679 $`

**References**
`CVE: CVE-2014-0224`
`BID:67899`
`Other:`
`URL:http://www.securityfocus.com/bid/67899`
`URL:http://openssl.org/`

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Weak Cipher Suites`

... continues on next page ...

... continued from previous page ...

OID:1.3.6.1.4.1.25623.1.0.103440
Version used: $Revision: 4863 $

**References**
CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000
Other:
URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html
URL:https://bettercrypto.org/
URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/

Medium (CVSS: 4.3)
NVT: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection

**Summary**
It was possible to detect the usage of the deprecated SSLv2 and/or SSLv3 protocol on this system.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to use the known cryptographic flaws to eavesdrop the connection between clients and the service to get access to sensitive data transferred within the secured connection.

**Solution**
**Solution type:** Mitigation
It is recommended to disable the deprecated SSLv2 and/or SSLv3 protocols in favor of the TLSv1+ protocols. Please see the references for more information.

**Affected Software/OS**
All services providing an encrypted communication using the SSLv2 and/or SSLv3 protocols.

**Vulnerability Insight**
The SSLv2 and SSLv3 protocols containing known cryptographic flaws like:
- Padding Oracle On Downgraded Legacy Encryption (POODLE, CVE-2014-3566)
- Decrypting RSA with Obsolete and Weakened eNcryption (DROWN, CVE-2016-0800)

**Vulnerability Detection Method**
Check the used protocols of the services provided by this system.
Details: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection
OID:1.3.6.1.4.1.25623.1.0.111012
Version used: $Revision: 4686 $

**References**

... continues on next page ...

... continued from previous page ...

CVE: CVE-2016-0800, CVE-2014-3566
Other:
URL:https://www.enisa.europa.eu/activities/identity-and-trust/library/deliverables/algorithms-key-sizes-and-parameters-report
URL:https://bettercrypto.org/
URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
URL:https://drownattack.com/
URL:https://www.imperialviolet.org/2014/10/14/poodle.html

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm
OID:1.3.6.1.4.1.25623.1.0.105880

... continues on next page ...

... continued from previous page ...

Version used: $Revision: 4781 $

**References**

```
Other:
  URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with
↪-sha-1-based-signature-algorithms/
```

[ return to 127.0.0.27 ]

### 2.31.3 Low 22/tcp

Low (CVSS: 2.6)
NVT: SSH Weak MAC Algorithms Supported

**Summary**
The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Disable the weak MAC algorithms.

**Vulnerability Detection Method**
Details: SSH Weak MAC Algorithms Supported
OID:1.3.6.1.4.1.25623.1.0.105610
Version used: $Revision: 4490 $

[ return to 127.0.0.27 ]

### 2.31.4 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

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

... continues on next page ...

... continued from previous page ...

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.27 ]

## 2.32 127.0.0.15

| | |
|---|---|
| Host scan start | Tue Feb 21 16:27:44 2017 UTC |
| Host scan end | Tue Feb 21 17:16:01 2017 UTC |

| Service (Port) | Threat Level |
|---|---|
| 22/tcp | Medium |
| 443/tcp | Medium |
| general/tcp | Low |
| 22/tcp | Low |

### 2.32.1 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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: 4490 $`

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

[ return to 127.0.0.15 ]

### 2.32.2 Medium 443/tcp

Medium (CVSS: 5.0)
NVT: SSL/TLS: Report Vulnerable Cipher Suites for HTTPS

**Summary**
This routine reports all SSL/TLS cipher suites accepted by a service where attack vectors exists only on HTTPS services.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

. . . continues on next page . . .

... continued from previous page ...

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Affected Software/OS**
Services accepting vulnerable SSL/TLS cipher suites via HTTPS.

**Vulnerability Insight**
These rules are applied for the evaluation of the vulnerable cipher suites:
- 64-bit block cipher 3DES vulnerable to the SWEET32 attack (CVE-2016-2183).

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Vulnerable Cipher Suites for HTTPS`
OID:1.3.6.1.4.1.25623.1.0.108031
Version used: `$Revision: 5232 $`

**References**
```
CVE: CVE-2016-2183, CVE-2016-6329
Other:
  URL:https://bettercrypto.org/
   URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
   URL:https://sweet32.info/
```

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:

... continues on next page ...

... continued from previous page ...

- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Weak Cipher Suites`
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: `$Revision: 4863 $`

**References**
`CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000`
`Other:`
`URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html`
`URL:https://bettercrypto.org/`
`URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/`

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.

... continues on next page ...

... continued from previous page ...

NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: `SSL/TLS: Certificate Signed Using A Weak Signature Algorithm`
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: `$Revision: 4781 $`

**References**
```
Other:
  URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with
↪-sha-1-based-signature-algorithms/
```

[ return to 127.0.0.15 ]

### 2.32.3 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**Affected Software/OS**

... continues on next page ...

... continued from previous page ...

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: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.15 ]

### 2.32.4 Low 22/tcp

Low (CVSS: 2.6)
NVT: SSH Weak MAC Algorithms Supported

**Summary**
The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Disable the weak MAC algorithms.

**Vulnerability Detection Method**
Details: `SSH Weak MAC Algorithms Supported`
OID:1.3.6.1.4.1.25623.1.0.105610
Version used: `$Revision: 4490 $`

[ return to 127.0.0.15 ]

## 2.33 127.0.0.17

| | |
|---|---|
| Host scan start | Tue Feb 21 15:25:05 2017 UTC |
| Host scan end | Tue Feb 21 16:28:17 2017 UTC |

| Service (Port) | Threat Level |
|---|---|
| 135/tcp | Medium |
| 3389/tcp | Medium |
| general/tcp | Low |

### 2.33.1 Medium 135/tcp

Medium (CVSS: 5.0)
NVT: DCE/RPC and MSRPC Services Enumeration Reporting

**Summary**
Distributed Computing Environment / Remote Procedure Calls (DCE/RPC) or MSRPC services running on the remote host can be enumerated by connecting on port 135 and doing the appropriate queries.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker may use this fact to gain more knowledge about the remote host.

**Solution**
**Solution type:** Mitigation
Filter incoming traffic to this ports.

**Vulnerability Detection Method**
Details: `DCE/RPC and MSRPC Services Enumeration Reporting`
OID:1.3.6.1.4.1.25623.1.0.10736
Version used: `$Revision: 4998 $`

[ return to 127.0.0.17 ]

### 2.33.2 Medium 3389/tcp

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

. . . continues on next page . . .

... continued from previous page ...

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Weak Cipher Suites`
OID:1.3.6.1.4.1.25623.1.0.103440
Version used: `$Revision: 4863 $`

**References**
`CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000`
`Other:`
`URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html`
`URL:https://bettercrypto.org/`
`URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/`

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**

... continues on next page ...

... continued from previous page ...

The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: `SSL/TLS: Certificate Signed Using A Weak Signature Algorithm`
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: `$Revision: 4781 $`

**References**
```
Other:
  URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with
↪-sha-1-based-signature-algorithms/
```

[ return to 127.0.0.17 ]

### 2.33.3 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

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

**Solution**
**Solution type:** Mitigation

... continues on next page ...

... continued from previous page ...

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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.17 ]

## 2.34 127.0.0.19

Host scan start Tue Feb 21 15:24:56 2017 UTC
Host scan end Tue Feb 21 16:03:58 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 80/tcp | Medium |
| 22/tcp | Medium |
| general/tcp | Low |
| 22/tcp | Low |

### 2.34.1 Medium 80/tcp

Medium (CVSS: 5.0)
NVT: Acme thttpd and mini_httpd Terminal Escape Sequence in Logs Command Injection Vulnerability

... continues on next page ...

... continued from previous page ...

**Summary**
Acme 'thttpd' and 'mini_httpd' are prone to a command-injection vulnerability because they fail to adequately sanitize user-supplied input in logfiles.
Attackers can exploit this issue to execute arbitrary commands in a terminal.
This issue affects thttpd 2.25b and mini_httpd 1.19 other versions may also be affected.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Vulnerability Detection Method**
Details: `Acme thttpd and mini_httpd Terminal Escape Sequence in Logs Command Injection V.`
`↪..`
OID:1.3.6.1.4.1.25623.1.0.100447
Version used: `$Revision: 3207 $`

**References**
`CVE: CVE-2009-4490, CVE-2009-4491`
`BID:37714`
`Other:`
`URL:http://www.securityfocus.com/bid/37714`
`URL:http://www.acme.com/software/mini_httpd/`
`URL:http://www.acme.com/software/thttpd/`
`URL:http://www.securityfocus.com/archive/1/508830`

[ return to 127.0.0.19 ]

### 2.34.2 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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.

... continues on next page ...

... continued from previous page ...

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: 4490 $`

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

[ return to 127.0.0.19 ]

### 2.34.3 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

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

... continues on next page ...

... continued from previous page ...

**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: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.19 ]

### 2.34.4 Low 22/tcp

Low (CVSS: 2.6)
NVT: SSH Weak MAC Algorithms Supported

**Summary**
The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Disable the weak MAC algorithms.

**Vulnerability Detection Method**
Details: `SSH Weak MAC Algorithms Supported`
OID:1.3.6.1.4.1.25623.1.0.105610
Version used: `$Revision: 4490 $`

[ return to 127.0.0.19 ]

## 2.35 127.0.0.12

| | |
|---|---|
| Host scan start | Tue Feb 21 15:56:50 2017 UTC |
| Host scan end | Tue Feb 21 18:04:20 2017 UTC |

| Service (Port) | Threat Level |
|---|---|
| 22/tcp | Medium |
| general/tcp | Low |
| 22/tcp | Low |

### 2.35.1 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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: 4490 $`

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

[ return to 127.0.0.12 ]

### 2.35.2 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.12 ]

### 2.35.3 Low 22/tcp

Low (CVSS: 2.6)
NVT: SSH Weak MAC Algorithms Supported

... continues on next page ...

. . . continued from previous page . . .

**Summary**
The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Disable the weak MAC algorithms.

**Vulnerability Detection Method**
Details: `SSH Weak MAC Algorithms Supported`
OID:1.3.6.1.4.1.25623.1.0.105610
Version used: `$Revision: 4490 $`

[ return to 127.0.0.12 ]

## 2.36 127.0.0.42

Host scan start Tue Feb 21 15:25:02 2017 UTC
Host scan end Tue Feb 21 16:02:57 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 22/tcp | Medium |
| 443/tcp | Medium |
| 22/tcp | Low |

### 2.36.1 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Disable the weak encryption algorithms.

**Vulnerability Insight**

. . . continues on next page . . .

... continued from previous page ...

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: 4490 $`

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

[ return to 127.0.0.42 ]

### 2.36.2 Medium 443/tcp

Medium (CVSS: 4.3)
NVT: SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability (POODLE)

**Summary**
This host is prone to an information disclosure vulnerability.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Successful exploitation will allow a man-in-the-middle attackers gain access to the plain text data stream.

**Solution**
**Solution type:** Mitigation
Possible Mitigations are:
- Disable SSLv3
- Disable cipher suites supporting CBC cipher modes
- Enable TLS_FALLBACK_SCSV if the service is providing TLSv1.0+

**Vulnerability Insight**

... continues on next page ...

... continued from previous page ...

The flaw is due to the block cipher padding not being deterministic and not covered by the Message Authentication Code

**Vulnerability Detection Method**
Evaluate previous collected information about this service.
Details: SSL/TLS: SSLv3 Protocol CBC Cipher Suites Information Disclosure Vulnerability .
↪..
OID:1.3.6.1.4.1.25623.1.0.802087
Version used: $Revision: 4749 $

**References**
CVE: CVE-2014-3566
BID:70574
Other:
URL:https://www.openssl.org/~bodo/ssl-poodle.pdf
URL:https://www.imperialviolet.org/2014/10/14/poodle.html
URL:https://www.dfranke.us/posts/2014-10-14-how-poodle-happened.html
URL:http://googleonlinesecurity.blogspot.in/2014/10/this-poodle-bites-exploit
↪ing-ssl-30.html

Medium (CVSS: 4.3)
NVT: SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection

**Summary**
It was possible to detect the usage of the deprecated SSLv2 and/or SSLv3 protocol on this system.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to use the known cryptographic flaws to eavesdrop the connection between clients and the service to get access to sensitive data transferred within the secured connection.

**Solution**
**Solution type:** Mitigation
It is recommended to disable the deprecated SSLv2 and/or SSLv3 protocols in favor of the TLSv1+ protocols. Please see the references for more information.

**Affected Software/OS**
All services providing an encrypted communication using the SSLv2 and/or SSLv3 protocols.

**Vulnerability Insight**
The SSLv2 and SSLv3 protocols containing known cryptographic flaws like:
- Padding Oracle On Downgraded Legacy Encryption (POODLE, CVE-2014-3566)

... continues on next page ...

... continued from previous page ...

- Decrypting RSA with Obsolete and Weakened eNcryption (DROWN, CVE-2016-0800)

**Vulnerability Detection Method**
Check the used protocols of the services provided by this system.
Details: `SSL/TLS: Deprecated SSLv2 and SSLv3 Protocol Detection`
OID:1.3.6.1.4.1.25623.1.0.111012
Version used: `$Revision: 4686 $`

**References**
```
CVE: CVE-2016-0800, CVE-2014-3566
Other:
  URL:https://www.enisa.europa.eu/activities/identity-and-trust/library/delivera
↪bles/algorithms-key-sizes-and-parameters-report
   URL:https://bettercrypto.org/
   URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/
   URL:https://drownattack.com/
   URL:https://www.imperialviolet.org/2014/10/14/poodle.html
```

Medium (CVSS: 4.3)
NVT: SSL/TLS: Report Weak Cipher Suites

**Summary**
This routine reports all Weak SSL/TLS cipher suites accepted by a service.
NOTE: No severity for SMTP services with 'Opportunistic TLS' and weak cipher suites on port 25/tcp is reported. If too strong cipher suites are configured for this service the alternative would be to fall back to an even more insecure cleartext communication.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
The configuration of this services should be changed so that it does not accept the listed weak cipher suites anymore.
Please see the references for more resources supporting you with this task.

**Vulnerability Insight**
These rules are applied for the evaluation of the cryptographic strength:
- RC4 is considered to be weak (CVE-2013-2566, CVE-2015-2808).
- Ciphers using 64 bit or less are considered to be vulnerable to brute force methods and therefore considered as weak (CVE-2015-4000).
- 1024 bit RSA authentication is considered to be insecure and therefore as weak.
- Any cipher considered to be secure for only the next 10 years is considered as medium
- Any other cipher is considered as strong

**Vulnerability Detection Method**
Details: `SSL/TLS: Report Weak Cipher Suites`

... continues on next page ...

... continued from previous page ...

OID:1.3.6.1.4.1.25623.1.0.103440
Version used: $Revision: 4863 $

**References**
CVE: CVE-2013-2566, CVE-2015-2808, CVE-2015-4000
Other:
URL:https://www.bsi.bund.de/SharedDocs/Warnmeldungen/DE/CB/warnmeldung_cb-k16-1465_update_6.html
URL:https://bettercrypto.org/
URL:https://mozilla.github.io/server-side-tls/ssl-config-generator/

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

... continues on next page ...

... continued from previous page ...

OID:1.3.6.1.4.1.25623.1.0.105880
Version used: $Revision: 4781 $

**References**
Other:
URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with-sha-1-based-signature-algorithms/

[ return to 127.0.0.42 ]

### 2.36.3 Low 22/tcp

Low (CVSS: 2.6)
NVT: SSH Weak MAC Algorithms Supported

**Summary**
The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Disable the weak MAC algorithms.

**Vulnerability Detection Method**
Details: SSH Weak MAC Algorithms Supported
OID:1.3.6.1.4.1.25623.1.0.105610
Version used: $Revision: 4490 $

[ return to 127.0.0.42 ]

## 2.37 127.0.0.40

Host scan start Tue Feb 21 15:24:58 2017 UTC
Host scan end Tue Feb 21 17:56:34 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 22/tcp | Medium |
| general/tcp | Low |
| 22/tcp | Low |

### 2.37.1 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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: 4490 $`

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

[ return to 127.0.0.40 ]

### 2.37.2 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
. . . continues on next page . . .

... continued from previous page ...

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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $`

**References**
```
Other:
  URL:http://www.ietf.org/rfc/rfc1323.txt
```

[ return to 127.0.0.40 ]

### 2.37.3 Low 22/tcp

Low (CVSS: 2.6)
NVT: SSH Weak MAC Algorithms Supported

**Summary**
The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Disable the weak MAC algorithms.

... continues on next page ...

. . . continued from previous page . . .

**Vulnerability Detection Method**
Details: `SSH Weak MAC Algorithms Supported`
OID:1.3.6.1.4.1.25623.1.0.105610
Version used: `$Revision: 4490 $`

[ return to 127.0.0.40 ]

## 2.38 127.0.0.11

Host scan start Tue Feb 21 15:57:21 2017 UTC
Host scan end Tue Feb 21 16:42:28 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 22/tcp | Medium |
| general/tcp | Low |
| 22/tcp | Low |

### 2.38.1 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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`

. . . continues on next page . . .

... continued from previous page ...

OID:1.3.6.1.4.1.25623.1.0.105611
Version used: $Revision: 4490 $

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

[ return to 127.0.0.11 ]

### 2.38.2 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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

... continues on next page ...

... continued from previous page ...

Version used: `$Revision: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.11 ]

### 2.38.3 Low 22/tcp

Low (CVSS: 2.6)
NVT: SSH Weak MAC Algorithms Supported

**Summary**
The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Disable the weak MAC algorithms.

**Vulnerability Detection Method**
Details: `SSH Weak MAC Algorithms Supported`
OID:1.3.6.1.4.1.25623.1.0.105610
Version used: `$Revision: 4490 $`

[ return to 127.0.0.11 ]

## 2.39 127.0.0.37

Host scan start Tue Feb 21 15:55:39 2017 UTC
Host scan end Tue Feb 21 16:50:05 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 80/tcp | Medium |
| 22/tcp | Medium |
| 22/tcp | Low |
| general/tcp | Low |

### 2.39.1 Medium 80/tcp

Medium (CVSS: 4.3)
NVT: Apache Web Server ETag Header Information Disclosure Weakness

**Summary**
A weakness has been discovered in Apache web servers that are configured to use the FileETag directive.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
Exploitation of this issue may provide an attacker with information that may be used to launch further attacks against a target network.

**Solution**
OpenBSD has released a patch that addresses this issue. Inode numbers returned from the server are now encoded using a private hash to avoid the release of sensitive information.
Novell has released TID10090670 to advise users to apply the available workaround of disabling the directive in the configuration file for Apache releases on NetWare. Please see the attached Technical Information Document for further details.

**Vulnerability Detection Method**
Due to the way in which Apache generates ETag response headers, it may be possible for an attacker to obtain sensitive information regarding server files. Specifically, ETag header fields returned to a client contain the file's inode number.
Details: `Apache Web Server ETag Header Information Disclosure Weakness`
OID:1.3.6.1.4.1.25623.1.0.103122
Version used: `$Revision: 3022 $`

**References**
`CVE: CVE-2003-1418`
`BID:6939`
`Other:`
`URL:https://www.securityfocus.com/bid/6939`
`URL:http://httpd.apache.org/docs/mod/core.html#fileetag`
`URL:http://www.openbsd.org/errata32.html`
`URL:http://support.novell.com/docs/Tids/Solutions/10090670.html`

[ return to 127.0.0.37 ]

### 2.39.2 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

. . . continues on next page . . .

... continued from previous page ...

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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: 4490 $`

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

[ return to 127.0.0.37 ]

### 2.39.3 Low 22/tcp

Low (CVSS: 2.6)
NVT: SSH Weak MAC Algorithms Supported

**Summary**
The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Disable the weak MAC algorithms.

**Vulnerability Detection Method**

... continues on next page ...

... continued from previous page ...

Details: SSH Weak MAC Algorithms Supported
OID:1.3.6.1.4.1.25623.1.0.105610
Version used: $Revision: 4490 $

[ return to 127.0.0.37 ]

### 2.39.4 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $

**References**
Other:

... continues on next page ...

... continued from previous page ...

URL:http://www.ietf.org/rfc/rfc1323.txt

[ return to 127.0.0.37 ]

## 2.40 127.0.0.21

Host scan start Tue Feb 21 16:21:43 2017 UTC
Host scan end Tue Feb 21 17:59:33 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 22/tcp | Medium |
| general/tcp | Low |
| 22/tcp | Low |

### 2.40.1 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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: 4490 $`

**References**

... continues on next page ...

... continued from previous page ...

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

[ return to 127.0.0.21 ]

### 2.40.2 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $`

**References**
`Other:`

... continues on next page ...

... continued from previous page ...

URL:http://www.ietf.org/rfc/rfc1323.txt

[ return to 127.0.0.21 ]

### 2.40.3 Low 22/tcp

Low (CVSS: 2.6)
NVT: SSH Weak MAC Algorithms Supported

**Summary**
The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Disable the weak MAC algorithms.

**Vulnerability Detection Method**
Details: `SSH Weak MAC Algorithms Supported`
OID:1.3.6.1.4.1.25623.1.0.105610
Version used: `$Revision: 4490 $`

[ return to 127.0.0.21 ]

## 2.41 127.0.0.16

Host scan start Tue Feb 21 15:55:50 2017 UTC
Host scan end Tue Feb 21 16:23:22 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 22/tcp | Medium |
| 22/tcp | Low |
| general/tcp | Low |

### 2.41.1 Medium 22/tcp

Medium (CVSS: 4.3)
NVT: SSH Weak Encryption Algorithms Supported

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

... continues on next page ...

... continued from previous page ...

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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: 4490 $`

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

[ return to 127.0.0.16 ]

### 2.41.2 Low 22/tcp

Low (CVSS: 2.6)
NVT: SSH Weak MAC Algorithms Supported

**Summary**
The remote SSH server is configured to allow weak MD5 and/or 96-bit MAC algorithms.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Disable the weak MAC algorithms.

**Vulnerability Detection Method**

... continues on next page ...

... continued from previous page ...

Details: SSH Weak MAC Algorithms Supported
OID:1.3.6.1.4.1.25623.1.0.105610
Version used: $Revision: 4490 $

[ return to 127.0.0.16 ]

### 2.41.3 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $

**References**
Other:

... continues on next page ...

... continued from previous page ...

URL:http://www.ietf.org/rfc/rfc1323.txt

[ return to 127.0.0.16 ]

## 2.42 127.0.0.24

Host scan start Tue Feb 21 15:24:42 2017 UTC
Host scan end Tue Feb 21 15:51:39 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| 443/tcp | Medium |

### 2.42.1 Medium 443/tcp

Medium (CVSS: 4.0)
NVT: SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerability

**Summary**
The SSL/TLS service uses Diffie-Hellman groups with insufficient strength (key size < 2048).

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Impact**
An attacker might be able to decrypt the SSL/TLS communication offline.

**Solution**
**Solution type:** Workaround
Deploy (Ephemeral) Elliptic-Curve Diffie-Hellman (ECDHE) or use a 2048-bit or stronger Diffie-Hellman group. (see https://weakdh.org/sysadmin.html).
For Apache Web Servers: Beginning with version 2.4.7, mod_ssl will use DH parameters which include primes with lengths of more than 1024 bits.

**Vulnerability Insight**
The Diffie-Hellman group are some big numbers that are used as base for the DH computations. They can be, and often are, fixed. The security of the final secret depends on the size of these parameters. It was found that 512 and 768 bits to be weak, 1024 bits to be breakable by really powerful attackers like governments.

**Vulnerability Detection Method**
Checks the DHE temporary public key size.
Details: SSL/TLS: Diffie-Hellman Key Exchange Insufficient DH Group Strength Vulnerabili.
↪..
OID:1.3.6.1.4.1.25623.1.0.106223
Version used: $Revision: 4739 $

... continues on next page ...

... continued from previous page ...

**References**
Other:
URL:https://weakdh.org/
URL:https://weakdh.org/sysadmin.html

Medium (CVSS: 4.0)
NVT: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm

**Summary**
The remote service is using a SSL/TLS certificate in the certificate chain that has been signed using a cryptographically weak hashing algorithm.

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** Mitigation
Servers that use SSL/TLS certificates signed with a weak SHA-1, MD5, MD4 or MD2 hashing algorithm will need to obtain new SHA-2 signed SSL/TLS certificates to avoid web browser SSL/TLS certificate warnings.

**Vulnerability Insight**
The following hashing algorithms used for signing SSL/TLS certificates are considered cryptographically weak and not secure enough for ongoing use:
- Secure Hash Algorithm 1 (SHA-1)
- Message Digest 5 (MD5)
- Message Digest 4 (MD4)
- Message Digest 2 (MD2)
Beginning as late as January 2017 and as early as June 2016, browser developers such as Microsoft and Google will begin warning users when visiting web sites that use SHA-1 signed Secure Socket Layer (SSL) certificates.
NOTE: The script preference allows to set one or more custom SHA-1 fingerprints of CA certificates which are trusted by this routine. The fingerprints needs to be passed comma-separated and case-insensitive:
Fingerprint1
or
fingerprint1,Fingerprint2

**Vulnerability Detection Method**
Check which hashing algorithm was used to sign the remote SSL/TLS certificate.
Details: SSL/TLS: Certificate Signed Using A Weak Signature Algorithm
OID:1.3.6.1.4.1.25623.1.0.105880
Version used: $Revision: 4781 $

**References**
Other:

... continues on next page ...

... continued from previous page ...

URL:https://blog.mozilla.org/security/2014/09/23/phasing-out-certificates-with-sha-1-based-signature-algorithms/

[ return to 127.0.0.24 ]

## 2.43 127.0.0.45

Host scan start Tue Feb 21 15:24:38 2017 UTC
Host scan end Tue Feb 21 16:26:17 2017 UTC

| Service (Port) | Threat Level |
|---|---|
| general/tcp | Low |

### 2.43.1 Low general/tcp

Low (CVSS: 3.5)
NVT: VMSA-2016-003: VMware ESXi updates address a cross-site scripting issue (remote check)

**Summary**
VMware product updates address a critical glibc security vulnerability

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**Solution**
**Solution type:** VendorFix
Apply the missing patch(es).

**Affected Software/OS**
ESXi 6.0 without patch ESXi600-201611102-SG ESXi 5.5 without patch ESXi550-201612102-SG

**Vulnerability Insight**
The ESXi Host Client contains a vulnerability that may allow for stored cross-site scripting (XSS). The issue can be introduced by an attacker that has permission to manage virtual machines through ESXi Host Client or by tricking the vSphere administrator to import a specially crafted VM. The issue may be triggered on the system from where ESXi Host Client is used to manage the specially crafted VM.

**Vulnerability Detection Method**
Check the build number
Details: VMSA-2016-003: VMware ESXi updates address a cross-site scripting issue (remote...
OID:1.3.6.1.4.1.25623.1.0.140101
Version used: $Revision: 4945 $

... continues on next page ...

... continued from previous page ...

**References**
CVE: CVE-2016-7463
Other:
URL:http://www.vmware.com/security/advisories/VMSA-2016-0023.html

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $

**References**
Other:
URL:http://www.ietf.org/rfc/rfc1323.txt

[ return to 127.0.0.45 ]

## 2.44 127.0.0.30

| | |
|---|---|
| Host scan start | Tue Feb 21 15:35:31 2017 UTC |
| Host scan end | Tue Feb 21 16:31:13 2017 UTC |

| Service (Port) | Threat Level |
|---|---|
| general/tcp | Low |

### 2.44.1 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

**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: 5309 $`

**References**

. . . continues on next page . . .

... continued from previous page ...

```
Other:
  URL:http://www.ietf.org/rfc/rfc1323.txt
```

[ return to 127.0.0.30 ]

## 2.45 127.0.0.48

| | |
|---|---|
| Host scan start | Tue Feb 21 15:51:39 2017 UTC |
| Host scan end | Tue Feb 21 17:34:35 2017 UTC |

| Service (Port) | Threat Level |
|---|---|
| general/tcp | Low |

### 2.45.1 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

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

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

**Vulnerability Detection Method**

... continues on next page ...

... continued from previous page ...

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: 5309 $`

**References**
`Other:`
`URL:http://www.ietf.org/rfc/rfc1323.txt`

[ return to 127.0.0.48 ]

## 2.46 127.0.0.18

| | |
|---|---|
| Host scan start | Tue Feb 21 16:27:44 2017 UTC |
| Host scan end | Tue Feb 21 17:07:05 2017 UTC |

| Service (Port) | Threat Level |
|---|---|
| general/tcp | Low |

### 2.46.1 Low general/tcp

Low (CVSS: 2.6)
NVT: TCP timestamps

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

**Vulnerability Detection Result**
Vulnerability was detected according to the Vulnerability Detection Method.

**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 also: http://www.microsoft.com/en-us/download/details.aspx?id=9152

... continues on next page ...

... continued from previous page ...

**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: 5309 $`

**References**
```
Other:
  URL:http://www.ietf.org/rfc/rfc1323.txt
```

[ return to 127.0.0.18 ]