

Unit: CFC011023

Vulnerabilities Assessment

Student No: S12

Student Name: Ivan Wong Sen Loong

Trainer's Name: Kar Wei

Date of completion: 24-03-2024

Function:

check_tool()

```
49 check_weak_passv
50 check_weak_passv
51 choose_password_
52 choose_username
53 g_message [18]
54 arch_log [157]
55 archsploit_funct
56 uid_ip [69]
57
58 ##### START OF FUNCTIONS #####
59
60 # Check for required tools
61 function check_tool() {
62     if ! command -v $1 && /dev/null
63     then
64         echo "$1 could not be found"
65         read -p "Would you like to install $1? (y/n): " install
66         if [ "$install" == "y" ]; then
67             echo "Installing $1..."
68             sudo apt-get install $1
69         else
70             echo "Please install $1 manually and re-run the script."
71             exit
72         fi
73     fi
74 }
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

```
(kali@kali) [~/Desktop/project]
$ cd /Desktop/project
cd: no such file or directory: /Desktop/project

(kali@kali) [~/Desktop/project]
$ cd ~/Desktop/project

(kali@kali) [~/Desktop/project]
$ sudo bash script.sh
[2024-03-23 18:51:59] Script started.
[2024-03-23 18:53:05] ##### START: VULNER ASSESMENT TOOL SETUP #####
seclists could not be found
Would you like to install seclists? (y/n):
```

Function:

valid_ip()

```
68 # Input validation for IP address
69 function valid_ip() {
70     local ip=$1
71     local stat=1
72
73     if [[ $ip =~ ^[0-9]+\.[0-9]+\.[0-9]+\.[0-9]+$ ]]; then
74         OIFS=$IFS
75         IFS='.'
76         ip=($ip)
77         IFS=$OIFS
78         [[ ${ip[0]} -le 255 && ${ip[1]} -le 255 && ${ip[2]} -le 255 && ${ip[3]} -le 255 ]]
79         stat=$?
80     fi
81     return $stat
82 }
83
84 # Function to choose username
85 function choose_username() {
86     read -p "Enter username (1.msadmin, 2.user): " USERNAME
87     case $USERNAME in
88         1)
89             echo "msadmin"
90             break
91         2)
92             echo "user"
93             break
94         *)
95             echo "Invalid username. Please enter 1 or 2."
96             choose_username
97         &&
98     esac
99 }
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

```
(kali@kali) [~/Desktop/project]
$ sudo bash script.sh
[2024-03-23 19:37:50] Script started.
[2024-03-23 19:37:54] ##### START: VULNER ASSESMENT TOOL SETUP #####
Enter target IP address: 10.0...
Invalid IP address. Please enter a valid IP address: 10..50.134
Invalid IP address. Please enter a valid IP address:
```

Function:

choose_username()

```

# Function to choose username
function choose_username() {
  read -p "Enter username (1.msfdadmin, 2.user [1/2]): " USERNAME
  case $USERNAME in
    1) USERNAME="msfdadmin" ;;
    2) USERNAME="user" ;;
    *) echo "Invalid choice." && choose_username ;;
  esac
}

# Function to choose password list
function choose_password_list() {
  echo "Select the password list source:"
  echo "1. Use built-in password list"
  echo "2. Use custom password list"
  read -p "Enter your choice (1/2): " PASSWORD_SOURCE
}

5900/tcp open  vnc      VNC (protocol 3.3)
6000/tcp open  X11       (access denied)
6667/tcp open  irc       UnrealIRCd
8009/tcp open  ajp13     Apache Jserv (Protocol v1.3)
8180/tcp open  http      Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 00:0C:29:39:49:95 (VMware)
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.51 seconds
[2024-03-23 19:41:50] // END: BASIC VULNER - NMAP SCAN //
[2024-03-23 19:41:51] // START: BASIC VULNER - WEAK PASSWORD SCAN //
Enter username (1.msfdadmin, 2.user): 3
Invalid choice.
Enter username (1.msfdadmin, 2.user): 4
Invalid choice.
Enter username (1.msfdadmin, 2.user): 1
```

Function:

choose_password_list()

- option 1 would use a password list in seclists (/usr/share/seclists/Passwords/UserPassCombo-Jay.txt)
- option 2 use a custom password list (called password_list_s12.lst) which has been created when the script is run, it only consist of two words which is "msfadmin" and "user"

```
94 # Function to choose password list
95 function choose_password_list() {
96     echo "Select the password list source:"
97     echo "1. Use built-in password list"
98     echo "2. Use custom password list"
99     read -p "Enter your choice (1/2): " PASSWORD_SOURCE
100
101     case $PASSWORD_SOURCE in
102         1)
103             PASSWORD_LIST="/usr/share/seclists/Passwords/UserPassCombo-Jay.txt" ;;
104         2)
105             read -p "Enter path to custom password list (may input: password_list_s12.lst): " CUSTOM_PASSWORD_LIST
106             PASSWORD_LIST="$CUSTOM_PASSWORD_LIST" ;;
107         *)
108             echo "Invalid choice." && choose_password_list ;;
109     esac
110 }
111
```

```
6000/tcp open  X11      (access denied)
6667/tcp open  irc      UnrealIRCd
8009/tcp open  ajp13    Apache Jserv (Protocol v1.3)
8180/tcp open  http     Apache Tomcat/Coyote JSP engine 1.1
MAC Address: 00:0C:29:39:49:95 (VMware)
Service Info: Hosts: metasploitable.localdomain, irc.Metasploitable.LAN; OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 13.51 seconds
[2024-03-23 19:41:50] ////////////////////////////////////////////////// END: BASIC VULNER - NMAP SCAN ////////////////////////////////////////
[2024-03-23 19:41:51] ////////////////////////////////////////////////// START: BASIC VULNER - WEAK PASSWORD SCAN//////////////////////////////////////
Enter username (1.msffadmin, 2.user): 3
Invalid choice.
Enter username (1.msffadmin, 2.user): 4
Invalid choice.
Enter username (1.msffadmin, 2.user): 1
Select the password list source:
1. Use built-in password list
2. Use custom password list
Enter your choice (1/2): 2
Enter path to custom password list (may input: password_list_s12.lst): password_list_s12.lst
```

Function:

check_weak_passwords() & check_weak_passwords_custom()

```
check_tool [53] 112 # Function to check for weak passwords: for choose password list option 1
check_weak_passv 113 # Difference lies in uppercase 'P'
check_weak_passv 114 function check_weak_passwords() {
choose_password_ 115 while read -r line; do
choose_username 116     hydra -l $USERNAME -P $line -t 2 -I $1 $2
log_message [18] 117     if [ $? -ne 0 ]; then
search_log [157] 118         break
searchsploit_functi 119     fi
valid_ip [69] 120 done < $PASSWORD_LIST
121 }
122
123 # Function to check for weak passwords: for choose password list option 1
124 # Difference lies in lowercase 'p'
125 function check_weak_passwords_custom() {
126 while read -r line; do
127     hydra -l $USERNAME -p $line -t 2 -I $1 $2
128     if [ $? -ne 0 ]; then
129         break
130     fi
131 done < $PASSWORD_LIST

Status Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-03-23 19:48:58
[WARNING] the rdp module is experimental. Please test, report - and if possible, fix.
Compiler [DATA] max 1 task per 1 server, overall 1 task, 1 login try (l:1/p:1), ~1 try per task
Messages [DATA] attacking rdp://192.168.150.134:3389/
[3389][rdp] account on 192.168.150.134 might be valid but account not active for remote desktop: 1
Scribble 1 of 1 target completed, 0 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-03-23 19:48:59
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret s
anyway).

Terminal Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-03-23 19:49:03
[DATA] max 1 task per 1 server, overall 1 task, 1 login try (l:1/p:1), ~1 try per task
[DATA] attacking ftp://192.168.150.134:21/
[21][ftp] host: 192.168.150.134 login: msfadmin password: msfadmin
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-03-23 19:49:04
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret s
anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-03-23 19:49:07
[DATA] max 1 task per 1 server, overall 1 task, 1 login try (l:1/p:1), ~1 try per task
[DATA] attacking ftp://192.168.150.134:21/
```

Function:

searchsploit_function()

- It would first grep the **version info** from 'nmap -sV' scan result

```
133 # function to perform vulnerability assessment using searchsploit
134
135 function searchsploit_function() {
136     # Perform an nmap scan to get service versions
137     nmap_result=$(nmap -sV $Target_IP)
138     # Extract service versions from nmap result
139     port_info=$(echo "$nmap_result" | grep -oP '^d+/\w+\s+\K[^\s]+')
140     # Remove anything within brackets from port info
141     port_info=$(echo "$port_info" | awk '{print $3}') # Extracting the version column only
142
143     echo "Service Versions:"
144     while IFS= read -r line; do
145         if [ ! -z "$line" ]; then
146             echo "$line"
147             fi
148         done <<< "$port_info"
149
150     # Loop through each port info and search for exploits
151     while IFS= read -r line; do
152         searchsploit $line
153     done <<< "$port_info"
154 }
155
```

Functions

- check_tool [53]
- check_weak_passv
- check_weak_passv
- choose_password_
- choose_username
- log_message [18]
- search_log [157]
- searchsploit_functi
- valid_ip [69]

Status

- OpenSSH
- Linux
- Postfix

Compiler

- ISC
- Apache

essages

- 2
- Samba

Scribble

- Samba
- netkit-rsh
- GNU
- Metasploitable

terminal

- 2-4
- ProFTPD
- MySQL
- PostgreSQL
- VNC
- (access
- UnrealIRCd
- Apache
- Apache

- then run through each of the line into the searchsploit command to find its NSE

```
Oracle Weblogic Apache Connector - POST Buffer | windows/remote/18897.rb
PHP 5.4.3 - apache request headers Function B | windows/remote/19231.rb
RedHat Apache 2.0.40 - Directory Index Default | linux/remote/23296.txt
RedHat Linux 7.0 Apache - Remote Username Enum | linux/remote/21112.php
Webfroot Shoutbox < 2.32 (Apache) - Local File | linux/remote/34.pl
-----
Shellcodes: No Results
-----
Paper Title | Path
-----
Apache Camel Exploitation | docs/english/47074-apache-camel-
Apache Ghostcat CVE 2020-1938 - Paper | docs/english/49716-apache-ghostc
Apache HTTP Server 2.4.50 Path Traversal and | docs/english/50552-apache-http-s
Apache2 Web Server Hardening Article (Turkish | docs/english/48033-apache2-web-s
Exploiting Apache James Server 2.3.2 | docs/english/40123-exploiting-ap
Hardening & messing with win32 Apache Tomcat | english/12878-hardening-&-messin
Linux Hardening & Security (cP/WHM + Apache) | english/12906-linux-hardening-&-
-----
[2024-03-23 19:55:20] ////////////////////////////////// END: FULL VULNER - SEARCHSPLOIT TARGET IP //////////////////////////////////
```


Function:

search_log()

```
155 # Function to search for a term in the log file
156 function search_log() {
157     while true; do
158         read -p "Enter search term: " search_term
159         if [ -z "$search_term" ]; then
160             echo "Search term cannot be empty."
161             continue
162         fi
163         echo "Searching for: $search_term"
164         grep -i "$search_term" "$log_file"
165
166         read -p "Do you want to search again? (y/n): " search_again
167         case $search_again in
168             [Yy]* ) continue;;
169             [Nn]* ) break;;
170             * ) echo "Please answer y or n.";;
171         esac
172     done
173 }
174
175
176 [DATA] attacking telnet://192.168.150.134:23/
177 1 of 1 target completed, 0 valid password found
178 Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-03-23 19:57:25
179 [2024-03-23 19:57:25] ////////////////////////////////////////////////// END: FULL VULNER - WEAK PASSWORD SCAN////////////////////////////////////
180 [2024-03-23 19:57:26] ////////////////////////////////////////////////// END: FULL VULNER //////////////////////////////////////
181 Enter search term: password: msfadmin
182 Searching for: password: msfadmin
183 [3889][rdp] account on 192.168.150.134 might be valid but account not active for remote desktop: login: msfadmin password: msfadmin, continuing attacking the account.
184 [21][ftp] host: 192.168.150.134 login: msfadmin password: msfadmin
185 [3889][rdp] account on 192.168.150.134 might be valid but account not active for remote desktop: login: msfadmin password: msfadmin, continuing attacking the account.
186 [21][ftp] host: 192.168.150.134 login: msfadmin password: msfadmin
187 [3889][rdp] account on 192.168.150.134 might be valid but account not active for remote desktop: login: msfadmin password: msfadmin, continuing attacking the account.
188 [21][ftp] host: 192.168.150.134 login: msfadmin password: msfadmin
189 [3889][rdp] account on 192.168.150.134 might be valid but account not active for remote desktop: login: msfadmin password: msfadmin, continuing attacking the account.
190 [21][ftp] host: 192.168.150.134 login: msfadmin password: msfadmin
191 [3889][rdp] account on 192.168.150.134 might be valid but account not active for remote desktop: login: msfadmin password: msfadmin, continuing attacking the account.
192 [21][ftp] host: 192.168.150.134 login: msfadmin password: msfadmin
193 Enter search term: Searching for: password: msfadmin
194 Do you want to search again? (y/n):
```