



TSR
Test Summary
Report

NetGun

Versione	0.9
Data	11/02/2023
Destinatario	Professore Carmine Gravino
Presentato da	Carlo Colizzi, Giulio Incoronato, Antonio Mazzearella

Team Members

Nome	Informazioni di contatto
Carlo Colizzi	c.colizzi@studenti.unisa.it
Giulio Incoronato	g.incoronato2@studenti.unisa.it
Antonio Mazzearella	a.mazzearella5@studenti.unisa.it

Revision History

Data	Versione	Descrizione	Autori
02/02/2023	0.1	Stesura del documento e aggiunta di Team Members e Revision History	Tutto il Gruppo
02/02/2023	0.2	Aggiunta del paragrafo 1, 2, 3, 4	Giulio Incoronato
05/02/2023	0.3	Aggiunta del paragrafo 5 e del sotto paragrafo "Filtering"	Giulio Incoronato
05/02/2023	0.4	Aggiunta del sotto paragrafo "Deep Scan"	Antonio Mazzarella
05/02/2023	0.5	Aggiunta del sotto paragrafo "Report"	Carlo Colizzi
06/02/2023	0.6	Creata una nuova tabella nel sottoparagrafo "Report"	Carlo Colizzi
07/02/2023	0.7	Create una nuova tabella nel sottoparagrafo "Filtering"	Giulio Incoronato
09/02/2023	0.8	Creata una nuova tabella nel sottoparagrafo "Deep Scan"	Antonio Mazzarella
11/02/2023	0.9	Inserito paragrafo 3 e aggiunto le tabelle "Riepilogo Generale", "Filtering", "Deep Scan", "Report"	Giulio Incoronato, Carlo Colizzi

Sommario

Team Members	1
Revision History	2
1 Introduzione	4
1.1 Scopo del documento	4
2 Relazioni con altri documenti	4
3 Risultati Test d'Unità	5
3.1 Filtering	5
3.2 Selezionamento Scan Mode	6
3.3 Inserimento path per esportazione del File di Report	7
4 Risultati Test di Sistema	8
4.1 Riepilogo Generale	8
4.2 Filtering	8
4.3 Deep Scan	9
4.4 Report	9

1 Introduzione

Nel mercato attuale sono presenti diversi software di questo tipo, che richiedono di sottoscrivere un abbonamento per utilizzarli e non sono personalizzabili in quanto sistemi Closed-Source.

Il sistema **NetGun** offre a tutti la possibilità di avere un tool per l'enumerazione dei servizi di rete, altamente personalizzabile e gratuito.

1.1 Scopo del documento

Lo scopo del documento è quello di fornire una visione generale di tutti quelli che sono i problemi riscontrati e non. In particolare nelle successive sezioni faremo un riepilogo di tutto ciò che è andato bene e di tutto ciò che non è andato bene, con annessa descrizione di come è stato risolto il relativo problema

2 Relazioni con altri documenti

Di seguito la relazione di tale documento con gli altri documenti di testing.

Test Plan

Nel Test Plan sono state specificate le modalità di esecuzione dei test da cui deriva tale report. Oltre ciò, nel Plan sono stati inseriti i test Frame del Category Partition per l'esecuzione dei test da cui deriva tale report.

Test Case Specification

Nel seguente report sono riportate le specifiche dei test case.

Test Incident Report

Un Test Incident Report è stato definito prima di questo documento e consiste in un report dell'esito dell'esecuzione dei test di sistema progettati nel Test Plan e nel Test Case specification

3 Risultati Test d'Unità

3.1 Filtering

Di seguito l'elenco dei test di unità andati a buon fine:

- TU_1.1_01_1
- TU_1.1_03_1
- TU_1.1_04_1
- TU_1.1_05_1
- TU_1.1_07_1
- TU_1.1_02_2
- TU_1.1_06_2

```
===== test session starts =====
collecting ... collected 8 items

test_checkipandport.py::Test_check_ip_and_port::test_long_ip_lenght PASSED [ 12%]
test_checkipandport.py::Test_check_ip_and_port::test_strange_ip_ports PASSED [ 25%]
test_checkipandport.py::Test_check_ip_and_port::test_normal_ip_ports PASSED [ 37%]
test_checkipandport.py::Test_check_ip_and_port::test_strange_ports_range PASSED [ 50%]
test_checkipandport.py::Test_check_ip_and_port::test_strange_ip_lenght PASSED [ 62%]
test_checkipandport.py::Test_check_ip_and_port::test_strange_everything PASSED [ 75%]
test_checkipandport.py::Test_check_ip_and_port::test_incorrect_format_ip PASSED [ 87%]
test_checkipandport.py::Test_check_ip_and_port::test_incorrect_everything PASSED [100%]

===== 8 passed in 0.01s =====
```

Di seguito l'elenco dei test di unità non andati a buon fine, risolti e con annessa descrizione:

- **TC_1.1_02_1**: L'errore è stato causato da un controllo non corretto dell'indirizzo IP, la funzione controllava solo il formato e non la lunghezza. In poche parole veniva restituito dalla funzione **check_ip()** sempre lo stesso valore (True). Si è risolto andando a modificare la variabile "pattern", attraverso una **regular expression**.
- **TC_1.1_06_1**: L'errore è stato causato da un controllo non corretto del range di porte, la funzione controllava solo il formato e non la lunghezza. Il metodo **check_ports()** restituiva lo stesso valore (True). Si è risolto modificando la variabile "pattern", attraverso una **regular expression**.

3.2 Selezionamento Scan Mode

Di seguito l'elenco dei test di unità andati a buon fine:

- TU_1.2_02_1
- TU_1.2_01_2

```
===== test session starts =====
collecting ... collected 2 items

test_deepscanning.py::Test_deep_scanning::test_if_no_input PASSED      [ 50%]
test_deepscanning.py::Test_deep_scanning::test_if_input_deep PASSED    [100%]

===== 2 passed, 3 warnings in 0.01s =====
```

Di seguito l'elenco dei test di unità non andati a buon fine, risolti e con annessa descrizione:

- **TU_1.2_01_1**: L'errore è stato causato da un controllo assente sulla variabile **scan_mode**. Non passando il parametro indicante la modalità di scan, la funzione restituiva un'eccezione **TypeError** non gestita. L'errore è stato risolto impostando un valore di default **"SHALLOW"** al parametro.

3.3 Inserimento path per esportazione del File di Report

- TU_1.3_01_1
- TU_1.3_02_1
- TU_1.3_03_2
- TU_1.3_04_2

```
===== test session starts =====
collecting ... collected 4 items

test_create_report.py::Test_create_report::test_max_lenght_path PASSED [ 25%]
test_create_report.py::Test_create_report::test_standard_path PASSED [ 50%]
test_create_report.py::Test_create_report::test_multiple_slash PASSED [ 75%]
test_create_report.py::Test_create_report::test_inverted_slash PASSED [100%]

===== 4 passed, 5 warnings in 0.22s =====
```

Di seguito l'elenco dei test di unità non andati a buon fine, risolti e con annessa descrizione:

- **TU_1.3_03_1:** L'errore è stato causato da un controllo non corretto della variabile path. La variabile passava il check della funzione anche se il formato non era corretto, andando a creare problemi successivamente nell'esecuzione della funzionalità. È stato risolto attraverso delle regular expression, utilizzate per controllare la validità del parametro, il tutto incapsulato nel metodo **check_path()**.
- **TU_1.3_04_1:** L'errore è stato causato dalle stesse problematiche descritte nel **TU_1.3_03_1**. E' stato risolto attraverso delle regular expression utilizzate nel metodo **check_path()**.

4 Risultati Test di Sistema

Per quanto riguarda il test di sistema sono state definite varie test suite svolte manualmente in un ambiente di testing virtualizzato. Nello specifico si è utilizzata una **Virtual Machine** hostata nella rete locale, per simulare la macchina target.

4.1 Riepilogo Generale

Esecuzione	Fallimenti	Successi
Esecuzione 1 05/02/2023	2	6
Esecuzione 2 09/02/2023	0	8

4.2 Filtering

Esecuzione	Fallimenti	Successi
Esecuzione 1 05/02/2023	2	6
Esecuzione 2 07/02/2023	0	8

4.3 Deep Scan

Esecuzione	Fallimenti	Successi
Esecuzione 1 05/02/2023	1	1
Esecuzione 2 09/02/2023	0	2

4.4 Report

Esecuzione	Fallimenti	Successi
Esecuzione 1 05/02/2023	2	2
Esecuzione 2 06/02/2023	0	4