

Documentation script de test SAE S4.Deploi.01

Tristan Petit, Nils Hubert, Toni Rey,
Majd El Sebeiti , Vianney Miquel

24 mars 2025



Table des matières

1	Introduction	1
2	Utilisation des scripts	2
2.1	Utilisation du script dans un terminal	2
2.2	Implémentation de cette documentation	2
2.3	Expression régulière	2
3	Test ping intranet	3
4	Test communication HTTP/HTTPS	4
5	Test de resolution de nom DNS	6

1 Introduction

Les scripts présentés dans ce document ont pour but de tester l'infrastructure réseaux développée dans le cadre de ce projet. Ces tests incluent :

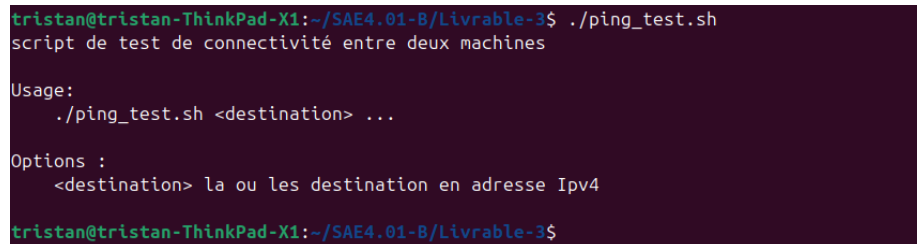
1. Test de connectivité de type ping
2. Test de connectivité de type DNS
3. The text in the entries may be of any length.

Pour certaines parties communes au script, dans un souci de non redondance la partie utilisation des scripts avec la doc dédiée possède une section spécifique.

2 Utilisation des scripts

Tous les scripts développés dans le cadre de ce projet possède une documentation, qui est l'équivalent d'un manuel. Pour y avoir accès il suffit de taper le nom du script tout seul et une documentation s'affichera.

2.1 Utilisation du script dans un terminal



```
tristan@tristan-ThinkPad-X1:~/SAE4.01-B/Livrable-3$ ./ping_test.sh
script de test de connectivité entre deux machines

Usage:
  ./ping_test.sh <destination> ...

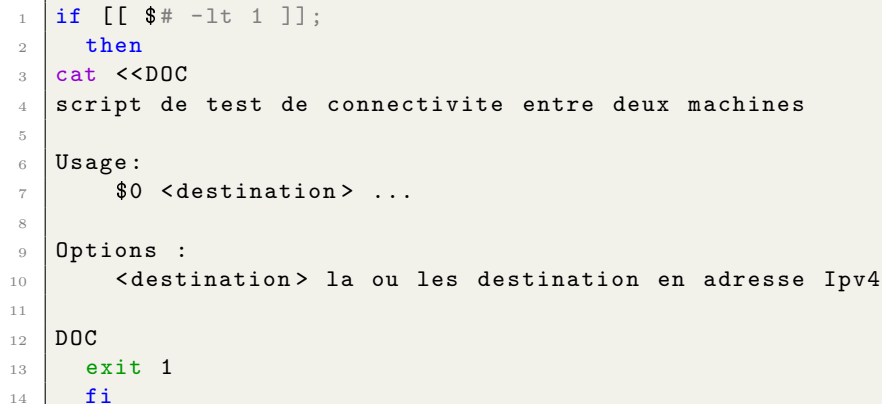
Options :
  <destination> la ou les destination en adresse Ipv4

tristan@tristan-ThinkPad-X1:~/SAE4.01-B/Livrable-3$
```

FIGURE 1 – Manuel d'utilisation du script

2.2 Implémentation de cette documentation

Pour cette documentation, nous avons inclus un if pour détecter si aucun paramètre n'a été donné au script. Dans ces cas là il lève une erreur en montrant les options que peut utiliser l'administrateur pour ce script.



```
1 if [[ $# -lt 1 ]];
2     then
3     cat <<DOC
4     script de test de connectivite entre deux machines
5
6     Usage:
7         $0 <destination> ...
8
9     Options :
10        <destination> la ou les destination en adresse Ipv4
11
12     DOC
13     exit 1
14     fi
```

Listing 1 – test HTTP/HTTPS

2.3 Expression régulière

Afin de faciliter la gestion des erreurs par les utilisateurs du script, nous avons mis en place une vérification des paramètres données par l'utilisateur via

les regex. Ces regex permettent de vérifier les entrées utilisateurs et de vérifier que celle ci soient valides et n'engendre pas de bug.

```
1 # Verification des adresses IPv4
2 regex_ip='^([0-9]{1,3}\.){3}[0-9]{1,3}$'
```

Listing 2 – expression régulière pour des adresses IP

3 Test ping intranet

Le but de ce script est de tester la connectivité entre les machines d'un réseau. Certaines machines ne seront autorisées à ping que certaines machines.

```
1 #!/bin/bash
2
3 #on teste la connectivité des machines entre elles
4
5 #test si les paramètres donnés sont valides
6
7 if [[ $# -lt 1 ]];
8 then
9 cat <<DOC
10 script de test de connectivité entre deux machines
11
12 Usage:
13 $0 <destination> ...
14
15 Options :
16 <destination> la ou les destination en adresse IPv4
17
18 DOC
19 exit 1
20 fi
21
22 # Verification des adresses IPv4
23 regex_ip='^([0-9]{1,3}\.){3}[0-9]{1,3}$'
24 numArg=1
25
26 for adresse in "$@";
27 do
28     if ! [[ $adresse =~ $regex_ip ]]; then
29         echo "Argument_$numArg_$adresse_IP_invalide:_"
30             $adresse"
31         exit 2
32     fi
33     (( numArg++ ))
34 done
35
36 for ip_adress in "$@";
```

```

36 do
37
38     if ping -w 1 "$ip_address" &> /dev/null;
39     then
40         echo "$USER communique correctement avec $ip_address"
41     else
42         echo "$USER n'arrive pas à communiquer avec $ip_address"
43     fi
44
45 done

```

Listing 3 – test ping

4 Test communication HTTP/HTTPS

Ce script http, permet de communiquer avec des sites web qui ont le port 80 d'ouvert.

```

1  #!/bin/bash
2
3  # si l'utilisateur n'entre pas de destination on lui affiche
   une documentation
4  if [[ $# -lt 1 ]];
5  then
6  cat <<DOC
7  script de test http sur le port 80
8
9  Usage :
10     $0 <destination> ...
11
12  Options :
13     <destination> hostname or Ip adress
14
15  entrez des nom de domaines ou des adresses ip valides !
16  DOC
17     exit 1
18  fi
19
20  # v rification du nom de domaine pass au script ou de l'
   adresse IP
21  for websiteCheck in "$@";
22  do
23     # check de l'adresse Ip et si ce n'est pas une adresse
       Ip on v rifie qu'il s'agit d'un nom de domaine
       valide
24     if [[ ! $websiteCheck =~ ^([0-9]{1,3}\.){3}[0-9]{1,3}$
       ]] && [[ ! $websiteCheck =~ ^[a-z][a-z]*\.[a-z][a-z]*[a-z]*\.[a-z]*$ ]];

```

```

25     then
26         echo "Le nom de domaine ou l'adresse Ip est invalide :
           $websiteCheck"
27         exit 1
28     fi
29 done
30
31 # on parcourt les noms de domaines donn es en param tres
    pour le HTTP
32 echo "
           ***** "
33 echo "***** TEST DE CONNECTION EN HTTP
           ***** "
34 echo "
           ***** "
35 for website in "$@";
36 do
37     # utilisation du logiciel netcat
38     if nc -w 1 "$website" 80;
39     then
40         echo "La communication avec le site web : $website
           fonctionne !"
41     else
42         echo "La communication avec le site web : $website
           ne fonctionne pas !"
43     fi
44 done
45 echo "
           ***** "
46 echo "***** FIN DU TEST DE CONNECTION EN HTTP
           ***** "
47 echo -e "
           ***** \
           n\n"
48
49 # test de la connection en HTTPS
50 echo "
           ***** "
51 echo "***** TEST DE CONNECTION EN HTTPS
           ***** "
52 echo "
           ***** "
53 for website in "$@";
54 do
55     # utilisation du logiciel netcat
56     if nc -w 1 "$website" 80;
57     then
58         echo "La communication avec le site web : $website
           fonctionne !"
59     else

```

```

60         echo "La communication avec le site web : $website
           ne fonctionne pas !"
61     fi
62 done
63 echo "
           ***** "
64 echo "***** FIN TEST DE CONNECTION EN HTTPS
           ***** "
65 echo "
           ***** "

```

Listing 4 – test HTTP/HTTPS

5 Test de resolution de nom DNS

Ici, le but est d'essayer la résolution de nom d'une VM et vérifier que celle ci fonctionne

```

1  #!/bin/bash
2
3  if [[ $# -lt 1 ]];
4  then
5      cat <<DOC
6  Script pour test de r solution dns
7
8  Usage :
9      $0 <destination> ...
10
11 Option :
12     <destination> nom de domaine de l' h t e _ e x _ : _ w w w .
13     facebook.com , _ facebook.com , _ facebook.fr
14 DOC
15 _ _ exit _ 1
16 _ _ fi
17 # _ v r i f i c a t i o n _ d ' u n nom de domaine du type
18
19 for testHost in "$@";
20 do
21     if ! [[ $testHost =~ ^[a-z][a-z]*\.[a-z][a-z][a-z]*\.[a-z]*\.[a-z]*$ ]]; then
22         echo "Domaine invalide : $testHost"
23         exit 1
24     fi
25 done
26
27 # communications avec toutes les adresse DNS
28
29 for host in "$@";

```



```

30 do
31     if host "$host" &> /dev/null; then
32         echo "la_r solution_de_nom_de_:_$host_fonctionne"
33     else
34         echo "la_r solution_de_nom_de_:_$host_a_ ch ou "
35     fi
36 done

```

Listing 5 – test ping