

Comment configurer la prise en charge de TACACS+ sur Cache Engine

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Components Used](#)

[Conventions](#)

[Configuration](#)

[Diagramme du réseau](#)

[Configurer le Cache Engine pour la prise en charge TACACS+](#)

[Vérification](#)

[Dépannage des commandes](#)

[Informations connexes](#)

[Introduction](#)

Ce document décrit comment configurer la prise en charge TACACS+ (Terminal Access Controller Access Control System Plus) afin d'accéder à Cisco Cache Engine. Les instructions de ce document vous permettent de vous valider sur un serveur/base de données TACACS+ distant lorsque vous établissez une connexion Telnet avec le Cache Engine. Si le serveur n'inclut pas d'entrée pour votre ID utilisateur, il vérifie localement les informations d'accès valides.

[Conditions préalables](#)

[Conditions requises](#)

Aucune spécification déterminée n'est requise pour ce document.

[Components Used](#)

Les informations contenues dans ce document sont basées sur les versions de matériel et de logiciel suivantes :

- Cisco Cache Engine 505 dans un environnement de laboratoire avec des configurations effacées
- Logiciel Cisco Cache Engine Version 2.3.1
- CiscoSecure pour UNIX

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is

live, make sure that you understand the potential impact of any command.

[Conventions](#)

Reportez-vous aux [Conventions des conseils techniques de Cisco](#) pour plus d'informations sur les conventions du document.

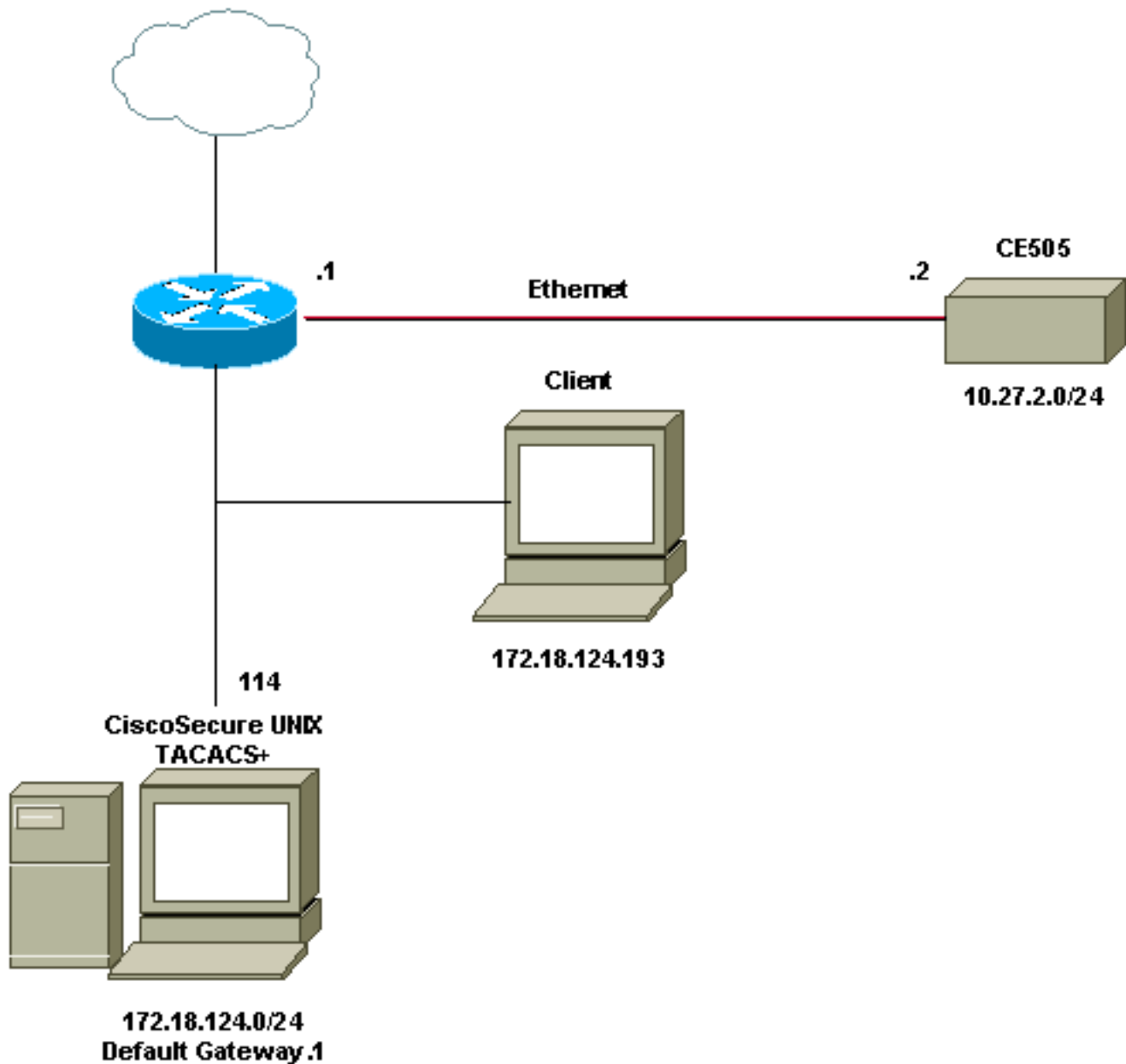
[Configuration](#)

Cette section vous fournit des informations pour configurer les fonctionnalités décrites dans ce document.

Remarque : utilisez l'[outil de recherche de commandes](#) (clients [enregistrés](#) uniquement) pour obtenir plus d'informations sur les commandes utilisées dans cette section.

[Diagramme du réseau](#)

Ce document utilise la configuration réseau suivante :



[Configurer le Cache Engine pour la prise en charge TACACS+](#)

Complétez ces étapes afin de configurer le Cache Engine pour la prise en charge TACACS+ :

1. Configurez le Cache Engine pour la version respective du protocole WCCP (Web Cache Communication Protocol).
2. Utilisez ces commandes pour la configuration par défaut :

```
authentication login local enable
authentication configuration local enable
```

3. Configurez l'adresse IP du serveur TACACS+. Si plusieurs serveurs indiquent quelle adresse est principale, les serveurs secondaires restent vides.
4. Configurez l'authentification sur le serveur TACACS+ en tant que serveur principal. Si le serveur n'est pas disponible, la valeur par défaut sera l'authentification spécifiée localement.
5. Configurez l'authentification sur les informations de clé TACACS+ si nécessaire.

Remarque : Vous devez activer TACACS+ sur Cisco Cache Engine, car Cisco Cache Engine utilise PPP pour s'authentifier auprès du serveur TACACS, contrairement aux routeurs qui ne

nécessitent pas PPP. Afin d'activer TACACS+ sur Cisco Cache Engine, ouvrez Cisco Secure ACS 2.6, cliquez sur l'onglet **Group Setup** et cochez la case **PPP IP** située dans la zone TACACS+ Settings.

Vos lignes de commande doivent apparaître comme suit :

```
cepro(config)#tacacs server 172.18.124.114
cepro(config)#authentication login tacacs ena primary
cepro(config)#authen configuration tacacs enab
```

Vérification

Référez-vous à cette section pour vous assurer du bon fonctionnement de votre configuration.

L'[Outil Interpréteur de sortie \(clients enregistrés uniquement\) \(OIT\) prend en charge certaines commandes show](#). Utilisez l'OIT pour afficher une analyse de la sortie de la commande **show**.

- **show version** : affiche le logiciel qui s'exécute sur le Cache Engine, ainsi que d'autres composants comme le temps de fonctionnement du système (par exemple, l'endroit où le code a été précédemment démarré et la date à laquelle il a été compilé).

```
cepro#show version
Cisco Cache Engine
Copyright (c) 1986-2001 by Cisco Systems, Inc.
Software Release: CE ver 2.31 (Build: FCS 02/16/01)
Compiled: 11:20:14 Feb 22 2001 by bbalagot
Image text-base 0x108000, data_base 0x437534

System restarted by Reload
The system has been up for 20 hours, 42 minutes, 59 seconds.
System booted from "flash"
```

- **show hardware** : affiche les mêmes informations que la commande **show version**, ainsi que les composants matériels du Cache Engine.

```
cepro#show hardware
Cisco Cache Engine
Copyright (c) 1986-2001 by Cisco Systems, Inc.
Software Release: CE ver 2.31 (Build: FCS 02/16/01)
Compiled: 11:20:14 Feb 22 2001 by bbalagot
Image text-base 0x108000, data_base 0x437534

System restarted by Reload
The system has been up for 21 hours, 15 minutes, 16 seconds.
System booted from "flash"
```

```
Cisco Cache Engine CE505 with CPU AMD-K6 (model 8) (rev. 12) AuthenticAMD
2 Ethernet/IEEE 802.3 interfaces
1 Console interface.
134213632 bytes of Physical Memory
131072 bytes of ROM memory.
8388608 bytes of flash memory.
```

```
List of disk drives:
/c0t0d0 (scsi bus 0, unit 0, lun 0)
```

- **show running-config** : affiche la configuration en cours sur le Cache Engine.

```
cepro#show running-config

Building configuration...
```

```

Current configuration:
!
!
!
user add admin uid 0 password 1 "eeSdy9dcy" capability admin-access
user add chbanks uid 5001 password 1 "eeSdy9dcy" capability admin-access
!
!
!
hostname ceopro
!
interface ethernet 0
 ip address 10.27.2.2 255.255.255.0
 ip broadcast-address 10.27.2.255
exit
!
!
interface ethernet 1
exit
!
ip default-gateway 10.27.2.1
ip route 0.0.0.0 0.0.0.0 10.27.2.1
cron file /local/etc/crontab
!
wccp router-list 1 10.27.2.1
wccp web-cache router-list-num 1
!
authentication login tacacs enable primary
authentication login local enable !--- on by default ---!
authentication configuration tacacs enable
authentication configuration local enable !---- on by default ---!
tacacs server 172.18.124.114 primary
rule no-cache url-regex .*cgi-bin.*
rule no-cache url-regex .*aw-cgi.*
!
!
end
cepro#

```

- **show tacacs** : affiche les paramètres du serveur TACACS+.

```

cepro#show tacacs
Login Authentication for Console/Telnet Session: enabled (primary)
Configuration Authentication for Console/Telnet Session: enabled

TACACS Configuration:
-----
Key          =
Timeout     = 5 seconds
Retransmit  = 2 times

Server                               Status
-----
172.18.124.114                       primary

```

- **show statistics tacacs** : affiche les statistiques TACACS+.

```

cepro#show statistics tacacs
TACACS+ Statistics
-----
Number of access requests: 13
Number of access deny responses: 7
Number of access allow responses: 0

```

- **show authentication** - Affiche la configuration actuelle d'authentification et d'autorisation TACACS+.

```
cepro#show authentication
Login Authentication:          Console/Telnet Session
-----
local                          enabled
tacacs                         enabled (primary)

Configuration Authentication: Console/Telnet Session
-----
local                          enabled
tacacs                         enabled

cepro#
```

Dépannage des commandes

Cette section fournit des informations que vous pouvez utiliser pour dépanner votre configuration.

L'[Outil Interpréteur de sortie \(clients enregistrés uniquement\) \(OIT\)](#) prend en charge certaines [commandes show](#). Utilisez l'OIT pour afficher une analyse de la sortie de la commande **show** .

Remarque : Consulter les [renseignements importants sur les commandes de débogage](#) avant d'utiliser les commandes de **débogage**.

- **show debug** : affiche les commandes debug activées.

```
cepro#show debug
Authentication debugging is on
Tacacs debugging is on
```

- **terminal monitor** : affiche les résultats des débogages à l'écran. Cette sortie affiche les résultats des commandes **debug authentication** et **debug tacacs**.

```
cepro#terminal monitor
cepro#authenticateUser(): Begin
setRemoteIPAddress(): pRemoteAddress 172.18.124.193
bAuthentication(): Begin
bAuthenticationIntersection(): Begin
bAuthenticationIntersection(): telnet_access 1
setAuthenticatedService(): nServiceToAuthenticate 6
getAuthenticatedService(): Begin
getAuthenticatedService(): nServiceToAuthenticate = 6
bAuthenticationIntersection() getAuthenticatedService 6
setErrorDisplayed(): Begin bStatus 0
getLocalLoginAuthEnable(): Begin
getLocalLoginAuthEnable(): uiState = 1
getTacacsLoginAuthEnable(): Begin
getTacacsLoginAuthEnable(): uiState = 1
getTacacsLoginAuthPrimary(): Begin
getTacacsLoginAuthPrimary(): uiState = 1
IncrementTacacsStatRequest(): Begin
tacacs_plus_login() Begin
isConsole() Begin
getAuthenticatedService(): Begin
getAuthenticatedService(): nServiceToAuthenticate = 6
isConsole() nReturn 0 telnet
tacacs_plus_login() sWhatService() tty = telnet
getRemoteIPAddress(): Begin
getRemoteIPAddress(): pRemoteAddress = 172.18.124.193
```

```

tacacs_plus_login() getRemoteIPAddress sHostIp 172.18.124.193
tacacs_malloc() Begin 164
tacacs_malloc() Pskmalloc ptr
getUserStruct() malloc_named ustr
tacacs_plus_login() allocated memory for ustruct
aaa_update_user() Begin
debug_authen_svc() Begin

aaa_update_user(): user='admin' ruser='system' port='telnet'
    rem_addr='172.18.124.193' authen_type=1
tacacs_plus_login() updated user
getNumTacacsLoginAttempts(): Begin
getNumTacacsLoginAttempts(): ulRetransmit = 2
##### tacacs_plus_login() num_tries 1
aaa_start_login() Begin
debug_start_login() Begin

debug_start_login()/AUTHEN/START (0): port='telnet' list='(null)'
    action=LOGIN service=LOGIN
aaa_randomize_id() Begin
tacacs_plus_start_login() Begin
tacacs_parse_server() Begin user_str admin
getTacacsDirectRequestEnable(): Begin
getTacacsDirectRequestEnable(): cDirectRequestEnable = 0
printIpAddr() Begin
printIpAddr() 0.0.0.0
tacacs_plus_start_login() server.ip_addr 0.0.0.0          server.type
    0 server.length 0
choose_version() Begin
create_authen_start() Begin
create_authen_start() len 45
tacacs_malloc() Begin 45
tacacs_malloc() Pskmalloc ptr
create_authen_start() malloc_named tac_pak
fill_tacacs_plus_hdr() Begin encrypt 1
fill_tacacs_plus_hdr() len 33, tac_pak->length 33
#### fill_tacacs_plus_hdr() tac_pak->encrypted 1
#### fill_tacacs_plus_hdr() TEST nTestLen 33
create_authen_start() len 33, tac_pak->length 33
create_authen_start() u->priv_lvl 15 start->priv_lvl 15
create_authen_start() start->action 1
create_authen_start() start->authen_type 1
create_authen_start() start->service 1
create_authen_start() user_len 5
create_authen_start() port_len 6
create_authen_start() addr_len 14
create_authen_start() out_len 33
tacacs_plus_start_login() TACACS+: send AUTHEN/START packet ver=192
    id=1541646967
tacacs_plus_start_login() login to TACACS+ server:
printIpAddr() Begin
printIpAddr() 0.0.0.0
tacacs_plus_get_conn() Begin server(0)
printIpAddr() Begin
printIpAddr() 0.0.0.0
tacacs_plus_get_conn() **pSocketHandleIndex 89434348
tacacs_plus_get_conn() Look at server in the TACACS+ server list
tacacs_plus_get_conn() TACACS+: This is a loop through server list
tacacs_plus_openconn() Begin
printIpAddr() Begin
printIpAddr() 172.18.124.114
open_handle() Begin
tacacs_plus_socket() Begin
tacacs_plus_socket Socket: return nSocket 784 nSockFdTbl[28] = 784

```

```
printIpAddress() Begin
printIpAddress() 172.18.124.114
open_handle() TACACS+: Opening TCP/IP connection to 172.18.124.114
open_handle() nSockFdTbl[28]= 784
setCurrentServer() Begin SaveCurrentServer->ip_addr 172.18.124.114
IncrementTacacsStatPerServerRequest(): Begin
##### IncrementTacacsStatPerServerRequest Server->ip_addr 1920733868
    tacacs_root.ulTacacsServerAddr
open_handle() socket(28) 784
tacacs_plus_connect() Begin
tacacs_plus_connect() socket(28) 784
tacacs_plus_connect() End
open_handle() is connected
open_handle() *connection_handle 28
open_handle() **pSocketHandleIndex 28
tacacs_plus_openconn() **pSocketHandleIndex 28
get_server() Begin
tacacs_plus_openconn() server->opens++
tacacs_plus_get_conn() **pSocketHandleIndex 28
tacacs_plus_get_conn() oldServerCount: 0, count:0
    tacacs_plus_start_login() **pHandleIndex 28
tacacs_plus_send_receive() Begin
tacacs_plus_proc_send_receive() Begin
tacacs_plus_proc_send_receive() length 33
copy_tac_plus_packet() Begin
tacacs_malloc() Begin 45
tacacs_malloc() PSkmalloc ptr
copy_tac_plus_packet() malloc_named copy
tacacs_plus_encrypt() Begin
getTacacsKey(): Begin
getTacacsKey(): sKey =
tacacs_plus_encrypt() key
tacacs_plus_encrypt() sizeof(tacacs_plus_pkt_hdr) 12
tacacs_plus_encrypt() sizeof(uchar) 1
tacacs_plus_encrypt() tac_pak->encrypted 1
tacacs_plus_encrypt() tac_pak->encrypted = TAC_PLUS_CLEAR && key is empty
tacacs_plus_proc_send_receive() out_pak->encrypted 1
tacacs_plus_proc_send_receive() out_pak->encrypted 1
tacacs_plus_proc_send_receive() PSkfree dump_pak
tacacs_plus_proc_send_receive() ntohl(out_pak->length) 33
dump_start_session() Begin ntohl(out_pak->length) 33
getTacacsKey(): Begin
getTacacsKey(): sKey =
0xc0 0x1 0x1 0x1 0x77 0xaa 0xe3 0x5b 0x0 0x0 0x0 0x21 0x1 0xf 0x1 0x1 0x5
    0x6 0xe 0x0 0x61 0x64 0x6d
encrypt_md5_xor() Begin
encrypt_md5_xor() no key
dump_summarise_incoming_packet_type() Begin
Read AUTHEN/START size=45
dump_nas_pak() Begin
dump_header() Begin
PACKET: key=
version 192 (0xc0), type 1, seq no 1, encrypted 1
session_id 2007688027 (0x77aae35b), Data length 33 (0x21)
End header
type=AUTHEN/START, priv_lvl = 15action=login
authen_type=ascii
service=login
user_len=5 port_len=6 (0x6), rem_addr_len=14 (0xe)
data_len=0
User: port: rem_addr: data:
End packet
dump_start_session() PSkfree test
getTacacsTimeout(): Begin
```



```
getTacacsTimeout(): ulTimeout = 5
tacacs_plus_sockwrite() Begin
tacacs_plus_proc_send_receive() PSkfree out_pak
getTacacsTimeout(): Begin
getTacacsTimeout(): ulTimeout = 5
sockread() Begin
tacacs_plus_proc_send_receive() read
tacacs_malloc() Begin 18
tacacs_malloc() PSkmalloc ptr
tacacs_plus_proc_send_receive() malloc_named *in
tacacs_plus_proc_send_receive() allocated memory
getTacacsTimeout(): Begin
getTacacsTimeout(): ulTimeout = 5
sockread() Begin
tacacs_plus_proc_send_receive() OK
tacacs_plus_decrypt() Begin
getTacacsKey(): Begin
getTacacsKey(): sKey =
tacacs_plus_decrypt() key
tacacs_plus_decrypt() tac_pak->encrypted = TAC_PLUS_CLEAR && key is empty
authen_resp_sanity_check() Begin
tacacs_plus_hdr_sanity_check() Begin
authen_debug_response() Begin
authen_debug_response() TACACS+: ver=192 id=1541646967 received AUTHEN
    status = FAIL
tacacs_plus_start_login() PSkfree out_tac_pak
unload_authen_resp() Begin
tacacs_plus_start_login() PSkfree in_tac_pak
debug_authen_status() Begin

TACACS+/AUTHEN (2007688027): status = FAIL

tacacs_plus_login() Authentication failed.
tacacs_plus_login() labell
aaa_cleanup_login() Begin
aaa_close_connection() Begin
tacacs_plus_closeconn() Begin
get_server() Begin
close_handle() Begin
close_handle() nHandleIndex 28 nSockFdTbl[**handle] 784
aaa_set_password() Begin
aaa_free_user() Begin
debug_authen_svc() Begin
aaa_close_connection() Begin

TACACS+/AUTHEN: free user admin system telnet 172.18.124.193
    authen_type=ASCII service=LOGIN priv_lv
aaa_free_user() PSkfree ustr
##### tacacs_plus_login() num_tries 2
aaa_start_login() Begin
debug_start_login() Begin

debug_start_login()/AUTHEN/START (0): port='unknown' list='(null)'
    action=LOGIN service=LOGIN

TACACS+/AUTHEN/START aaa_start_login() (0): ERROR (no ustruct)
    tacacs_plus_login() TACACS+: aaa_start
aaa_free_user() Begin
tacacs_plus_login() try_local_login AUTHENTICATION_INTERNAL_ERROR
IncrementTacacsStatDenyAccess(): Begin
localAuthentication(): Begin
localAuthentication() usrName admin
localAuthentication() passwd system
localAuthentication() pUid 89435294
```

```
localAuthentication() telnet_access
localAuthentication() rc == TRUE
AuthenticationIntersection(): bTacacsLogin 0
IncrementLocalLoginStat(): Begin
getLocalConfigAuthEnable(): Begin
getLocalConfigAuthEnable(): uiState = 1
getTacacsConfigAuthEnable(): Begin
getTacacsConfigAuthEnable(): uiState = 1
getTacacsConfigAuthPrimary(): Begin
getTacacsConfigAuthPrimary(): uiState = 0
localAuthentication(): Begin
localAuthentication() usrName admin
localAuthentication() passwd system
localAuthentication() pUid 89435294
localAuthentication() telnet_access
localAuthentication() rc == TRUE
AuthenticationIntersection(): bTacacsConfig 0
AuthenticationIntersection(): Local Database Authentication ==
IncrementLocalConfigStat(): Begin
AuthenticationIntersection(): user has been found
AuthenticationIntersection(): bTacacsLogin pUid 89435294
AuthenticationIntersection(): GOT ACCESS capab 0 Admin 0 Ftp 0 Http 0
    Telnet 0

authenticateUser() AUTHENTICATION IS OK
authenticateUser() AUTHENTICATION #2
```

[Informations connexes](#)

- [Produits et services de Cache Engine de la gamme Cisco 500](#)