

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Composants utilisés](#)

[Conventions](#)

[Configurez](#)

[Diagramme du réseau](#)

[Configurez le moteur de cache pour le support TACACS+](#)

[Vérifiez](#)

[Dépannage des commandes](#)

[Informations connexes](#)

[Introduction](#)

Ce document décrit comment configurer le support du Terminal Access Controller Access Control System Plus (TACACS+) afin d'accéder au moteur de cache de Cisco. Les instructions dans ce document te permettent pour valider contre un serveur/base de données du distant TACACS+ quand vous telnet au moteur de cache. Si le serveur n'inclut pas une entrée pour votre user-id, il vérifie localement les informations d'accès valides.

[Conditions préalables](#)

[Conditions requises](#)

Aucune spécification déterminée n'est requise pour ce document.

[Composants utilisés](#)

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

- Moteur de cache 505 de Cisco dans un environnement de travaux pratiques avec des configurations effacées
- Version de logiciel 2.3.1 de moteur de cache de Cisco
- CiscoSecure pour l'UNIX

Les informations contenues dans ce document ont été créées à partir des périphériques d'un environnement de laboratoire spécifique. Tous les périphériques utilisés dans ce document ont démarré avec une configuration effacée (par défaut). Si votre réseau est opérationnel, assurez-vous que vous comprenez l'effet potentiel de toute commande.

[Conventions](#)

Référez-vous au [Conventions relatives aux conseils techniques Cisco](#) pour les informations sur des conventions de document.

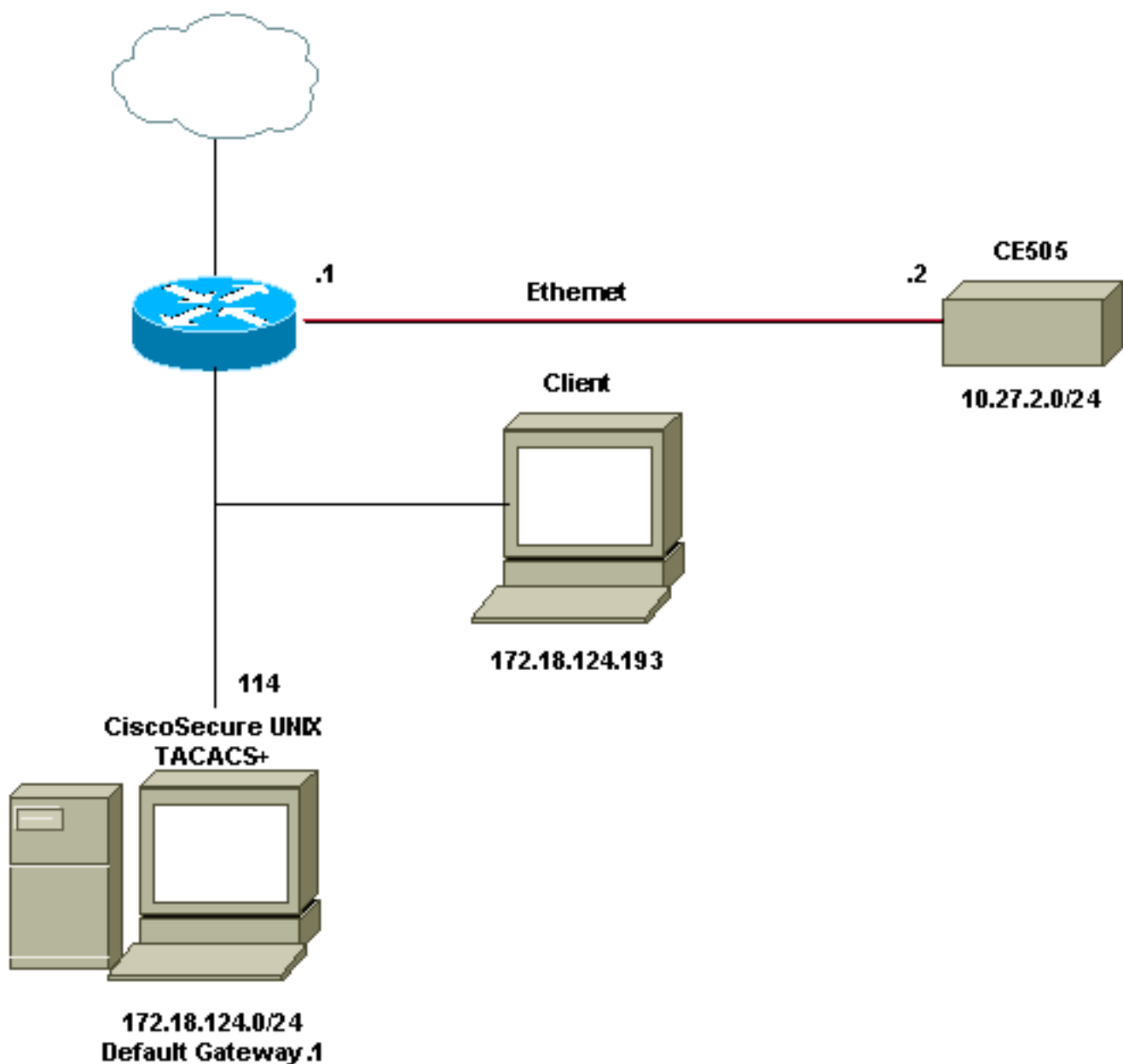
Configurez

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

Remarque: Utilisez l'outil [Command Lookup Tool](#) (clients [enregistrés](#) seulement) pour obtenir plus d'informations sur les commandes utilisées dans cette section.

Diagramme du réseau

Ce document utilise la configuration réseau suivante :



[Configurez le moteur de cache pour le support TACACS+](#)

Terminez-vous ces étapes afin de configurer le moteur de cache pour le support TACACS+ :

1. Configurez le moteur de cache pour la version respective du Web Cache Communication Protocol (WCCP).
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 les plusieurs serveurs spécifient que l'adresse est primaire, alors les serveurs secondaires sont laissés en tant qu'options vides.
4. Configurez l'authentification au serveur TACACS+ comme primaire. Si le serveur n'est pas disponible, alors le par défaut sera l'authentification localement spécifiée.
5. Configurez l'authentification à l'information principale TACACS+ en cas de besoin.

Remarque: Vous devez activer TACACS+ sur le moteur de cache de Cisco parce que les moteurs de cache de Cisco emploient le PPP afin d'authentifier avec le serveur TACACS, à la différence des Routeurs qui n'ont pas besoin du PPP. Afin d'activer TACACS+ sur les moteurs de cache de Cisco, le Cisco Secure ACS ouvert 2.6, cliquer sur l'onglet de **Group Setup**, et cocher la case **IP de PPP** située dans la région de configurations TACACS+.

Vos lignes de commande devraient ressembler à cette sortie :

```
cepro(config)#tacacs server 172.18.124.114cepro(config)#authentication login tacacs enable
cepro(config)#authen configuration tacacs enable
```

Vérifiez

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 fonctionne sur le moteur de cache, aussi bien que quelques autres composants comme disponibilité système (telle qu'où le code a été précédemment amorcé et la date où 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"
- **affichez le matériel ?** Affiche les mêmes informations que la commande de **show version**, aussi bien que les composants matériels du moteur de cache.
`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)
AuthenticAMD2 Ethernet/IEEE 802.3 interfaces1 Console interface.134213632 bytes of Physical Memory131072 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 moteur de cache.
`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 cepro!
interface ethernet 0 ip address 10.27.2.2 255.255.255.0 ip broadcast-address 10.27.2.255
exit!!
interface ethernet 1exit!
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 10.27.2.1wccp 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. *!!endcepro#
```

- **show tacacs ?** Affiche les configurations pour le 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
```
- **affichez les tacacs de statistiques ?** Statistiques des affichages 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 en cours d'authentification et d'autorisation du courant TACACS+.

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

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: Référez-vous aux [informations importantes sur les commandes de débogage](#) avant d'utiliser les commandes de **débogage**.

- **show debug ?** Affiche les commandes de débogage qui sont activées.

```
cepro#show
debug Authentication debugging is on Tacacs debugging is on
```
- **terminal monitor ?** Affiche les sorties de met au point à l'écran. Cette sortie affiche les résultats du **debug authentication** et des commandes de **debug tacacs**.

```
cepro#terminal
monitorcepro#authenticateUser(): BeginsetRemoteIPAddress(): pRemoteAddress
172.18.124.193bAuthentication(): BeginbAuthenticationIntersection():
BeginbAuthenticationIntersection(): telnet_access 1setAuthenticatedService():
nServiceToAuthenticate 6getAuthenticatedService(): BegingetAuthenticatedService():
nServiceToAuthenticate = 6bAuthenticationIntersection() getAuthenticatedService
6setErrorDisplayed(): Begin bStatus 0getLocalLoginAuthEnable():
BegingetLocalLoginAuthEnable(): uiState = 1getTacacsLoginAuthEnable():
BegingetTacacsLoginAuthEnable(): uiState = 1getTacacsLoginAuthPrimary():
BegingetTacacsLoginAuthPrimary(): uiState = 1IncrementTacacsStatRequest():
Begin tacacs_plus_login() BeginisConsole() BegingetAuthenticatedService():
BegingetAuthenticatedService(): nServiceToAuthenticate = 6isConsole() nReturn 0
telnettacacs_plus_login() sWhatService() tty = telnetgetRemoteIPAddress():
BegingetRemoteIPAddress(): pRemoteAddress = 172.18.124.193tacacs_plus_login()
getRemoteIPAddress sHostIp 172.18.124.193tacacs_malloc() Begin 164tacacs_malloc() Pskmalloc
ptrgetUserStruct() malloc_named ustrtacacs_plus_login() allocated memory for
ustructaaa_update_user() Begindebug_authen_svc() Beginaaa_update_user(): user='admin'
ruser='system' port='telnet' rem_addr='172.18.124.193' authen_type=1tacacs_plus_login()
updated usergetNumTacacsLoginAttempts(): BegingetNumTacacsLoginAttempts(): ulRetransmit =
2##### tacacs_plus_login() num_tries 1aaa_start_login() Begindebug_start_login()
Begindebug_start_login()/AUTHEN/START (0): port='telnet' list='(null)' action=LOGIN
service=LOGINaaa_randomize_id() Begin tacacs_plus_start_login() Begin tacacs_parse_server()
Begin user_str admingetTacacsDirectRequestEnable(): BegingetTacacsDirectRequestEnable():
cDirectRequestEnable = 0printIpAddr() BeginprintIpAddr() 0.0.0.0tacacs_plus_start_login()
server.ip_addr 0.0.0.0 server.type 0 server.length 0choose_version()
Begincreate_authen_start() Begincreate_authen_start() len 45tacacs_malloc() Begin
45tacacs_malloc() Pskmalloc ptrcreate_authen_start() malloc_named
tac_pakfill_tacacs_plus_hdr() Begin encrypt 1fill_tacacs_plus_hdr() len 33, tac_pak->length
33#### fill_tacacs_plus_hdr() tac_pak->encrypted 1#### fill_tacacs_plus_hdr() TEST nTestLen
```

```

33create_authen_start() len 33, tac_pak->length 33create_authen_start() u->priv_lvl 15
start->priv_lvl 15create_authen_start() start->action 1create_authen_start() start-
>authen_type 1create_authen_start() start->service 1create_authen_start() user_len
5create_authen_start() port_len 6create_authen_start() addr_len 14create_authen_start()
out_len 33tacacs_plus_start_login() TACACS+: send AUTHEN/START packet ver=192
id=1541646967tacacs_plus_start_login() login to TACACS+ server:printIpAddr()
BeginprintIpAddr() 0.0.0.0tacacs_plus_get_conn() Begin server(0)printIpAddr()
BeginprintIpAddr() 0.0.0.0tacacs_plus_get_conn() **pSocketHandleIndex
89434348tacacs_plus_get_conn() Look at server in the TACACS+ server
listtacacs_plus_get_conn() TACACS+: This is a loop through server listtacacs_plus_openconn()
BeginprintIpAddr() BeginprintIpAddr() 172.18.124.114open_handle() Begin tacacs_plus_socket()
Begin tacacs_plus_socket Socket: return nSocket 784 nSockFdTbl[28] = 784printIpAddr()
BeginprintIpAddr() 172.18.124.114open_handle() TACACS+: Opening TCP/IP connection to
172.18.124.114open_handle() nSockFdTbl[28]= 784setCurrentServer() Begin SaveCurrentServer-
>ip_addr 172.18.124.114IncrementTacacsStatPerServerRequest(): Begin####
IncrementTacacsStatPerServerRequest Server->ip_addr 1920733868
tacacs_root.ulTacacsServerAddropen_handle() socket(28) 784tacacs_plus_connect()
Begin tacacs_plus_connect() socket(28) 784tacacs_plus_connect() Endopen_handle() is
connectedopen_handle() *connection_handle 28open_handle() **pSocketHandleIndex
28tacacs_plus_openconn() **pSocketHandleIndex 28get_server() Begin tacacs_plus_openconn()
server->opens++tacacs_plus_get_conn() **pSocketHandleIndex 28tacacs_plus_get_conn()
oldServerCount: 0, count:0 tacacs_plus_start_login() **pHandleIndex
28tacacs_plus_send_receive() Begin tacacs_plus_proc_send_receive()
Begin tacacs_plus_proc_send_receive() length 33copy_tac_plus_packet() Begin tacacs_malloc()
Begin 45tacacs_malloc() PSkmalloc ptrcopy_tac_plus_packet() malloc_named
copytacacs_plus_encrypt() Begin getTacacsKey(): Begin getTacacsKey(): sKey
=tacacs_plus_encrypt() keytacacs_plus_encrypt() sizeof(tacacs_plus_pkt_hdr)
12tacacs_plus_encrypt() sizeof(uchar) 1tacacs_plus_encrypt() tac_pak->encrypted
1tacacs_plus_encrypt() tac_pak->encrypted = TAC_PLUS_CLEAR && key is
emptytacacs_plus_proc_send_receive() out_pak->encrypted 1tacacs_plus_proc_send_receive()
out_pak->encrypted 1tacacs_plus_proc_send_receive() PSkfree
dump_paktacacs_plus_proc_send_receive() ntohl(out_pak->length) 33dump_start_session() Begin
ntohl(out_pak->length) 33getTacacsKey(): 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
0x6dencrypt_md5_xor() Begin encrypt_md5_xor() no keydump_summarise_incoming_packet_type()
Begin Read AUTHEN/START size=45dump_nas_pak() Begin dump_header() Begin PACKET: key=version 192
(0xc0), type 1, seq no 1, encrypted 1session_id 2007688027 (0x77aae35b), Data length 33
(0x21)End headertype=AUTHEN/START, priv_lvl =
15action=loginauthen_type=asciiservice=loginuser_len=5 port_len=6 (0x6), rem_addr_len=14
(0xe)data_len=0User: port: rem_addr: data:End packetdump_start_session() PSkfree
testgetTacacsTimeout(): Begin getTacacsTimeout(): ulTimeout = 5tacacs_plus_sockwrite()
Begin tacacs_plus_proc_send_receive() PSkfree out_paketTacacsTimeout():
Begin getTacacsTimeout(): ulTimeout = 5sockread() Begin tacacs_plus_proc_send_receive()
readtacacs_malloc() Begin 18tacacs_malloc() PSkmalloc ptrtacacs_plus_proc_send_receive()
malloc_named *intacacs_plus_proc_send_receive() allocated memorygetTacacsTimeout():
Begin getTacacsTimeout(): ulTimeout = 5sockread() Begin tacacs_plus_proc_send_receive()
OKtacacs_plus_decrypt() Begin getTacacsKey(): Begin getTacacsKey(): sKey
=tacacs_plus_decrypt() keytacacs_plus_decrypt() tac_pak->encrypted = TAC_PLUS_CLEAR && key
is emptyauthen_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 = FAILtacacs_plus_start_login() PSkfree
out_tac_pakunload_authen_resp() Begin tacacs_plus_start_login() PSkfree
in_tac_pakdebug_authen_status() Begin TACACS+/AUTHEN (2007688027): status =
FAILtacacs_plus_login() Authentication failed.tacacs_plus_login() labellaaa_cleanup_login()
Begin aaa_close_connection() Begin tacacs_plus_closeconn() Begin get_server()
Begin close_handle() Begin close_handle() nHandleIndex 28 nSockFdTbl[**handle]
784aaa_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_lvaaa_free_user() PSkfree
ustr##### tacacs_plus_login() num_tries 2aaa_start_login() Begin debug_start_login()
Begin debug_start_login()/AUTHEN/START (0): port='unknown' list='(null)' action=LOGIN
service=LOGINTACACS+/AUTHEN/START aaa_start_login() (0): ERROR (no ustruct)
tacacs_plus_login() TACACS+: aaa_startaaa_free_user() Begin tacacs_plus_login()
try_local_login AUTHENTICATION_INTERNAL_ERRORIncrementTacacsStatDenyAccess():

```

```
BeginlocalAuthentication(): BeginlocalAuthentication() usrName adminlocalAuthentication()  
passwd systemlocalAuthentication() pUId 89435294localAuthentication()  
telnet_accesslocalAuthentication() rc == TRUEAuthenticationIntersection(): bTacacsLogin  
0IncrementLocalLoginStat(): BegingetLocalConfigAuthEnable():  
BegingetLocalConfigAuthEnable(): uiState = 1getTacacsConfigAuthEnable():  
BegingetTacacsConfigAuthEnable(): uiState = 1getTacacsConfigAuthPrimary():  
BegingetTacacsConfigAuthPrimary(): uiState = 0localAuthentication():  
BeginlocalAuthentication() usrName adminlocalAuthentication() passwd  
systemlocalAuthentication() pUId 89435294localAuthentication()  
telnet_accesslocalAuthentication() rc == TRUEAuthenticationIntersection(): bTacacsConfig  
0AuthenticationIntersection():= Local Database Authentication ==IncrementLocalConfigStat():  
BeginAuthenticationIntersection(): user has been foundAuthenticationIntersection():  
bTacacsLogin pUId 89435294AuthenticationIntersection(): GOT ACCESS capab 0 Admin 0 Ftp 0  
Http 0 Telnet 0authenticateUser() AUTHENTICATION IS OKauthenticateUser() AUTHENTICATION  
#2
```

[Informations connexes](#)

- [Notes de mise à jour pour le Logiciel de cache Cisco, version 2.3.0 - Mise en antémémoire de FTP](#)
- [Produits et services de moteur de cache de gamme Cisco 500](#)
- [Support et documentation techniques - Cisco Systems](#)