

Contenido

[Introducción](#)

[prerrequisitos](#)

[Requisitos](#)

[Componentes Utilizados](#)

[Convenciones](#)

[Configurar](#)

[Diagrama de la red](#)

[Configure el motor del caché para el soporte de TACACS+](#)

[Verificación](#)

[Comandos para resolución de problemas](#)

[Información Relacionada](#)

[Introducción](#)

Este documento describe cómo configurar el Terminal Access Controller Access Control System más el soporte (TACACS+) para acceder el Cisco Cache Engine. Las instrucciones en este documento permiten que usted valide contra un servidor/una base de datos del telecontrol TACACS+ cuando usted telnet al motor del caché. Si el servidor no incluye una entrada para su identificación del usuario, marca localmente para saber si hay la información de acceso válida.

[prerrequisitos](#)

[Requisitos](#)

No hay requisitos específicos para este documento.

[Componentes Utilizados](#)

La información que contiene este documento se basa en las siguientes versiones de software y hardware.

- Cisco Cache Engine 505 en un ambiente de laboratorio con las configuraciones despejadas
- Versión 2.3.1 del software Cisco Cache Engine.
- CiscoSecure para UNIX

La información que contiene este documento se creó a partir de los dispositivos en un ambiente de laboratorio específico. Todos los dispositivos que se utilizan en este documento se pusieron en funcionamiento con una configuración verificada (predeterminada). Si la red está funcionando, asegúrese de haber comprendido el impacto que puede tener cualquier comando.

[Convenciones](#)

Refiera a los [convenios de los consejos técnicos de Cisco](#) para la información sobre las

convenciones sobre documentos.

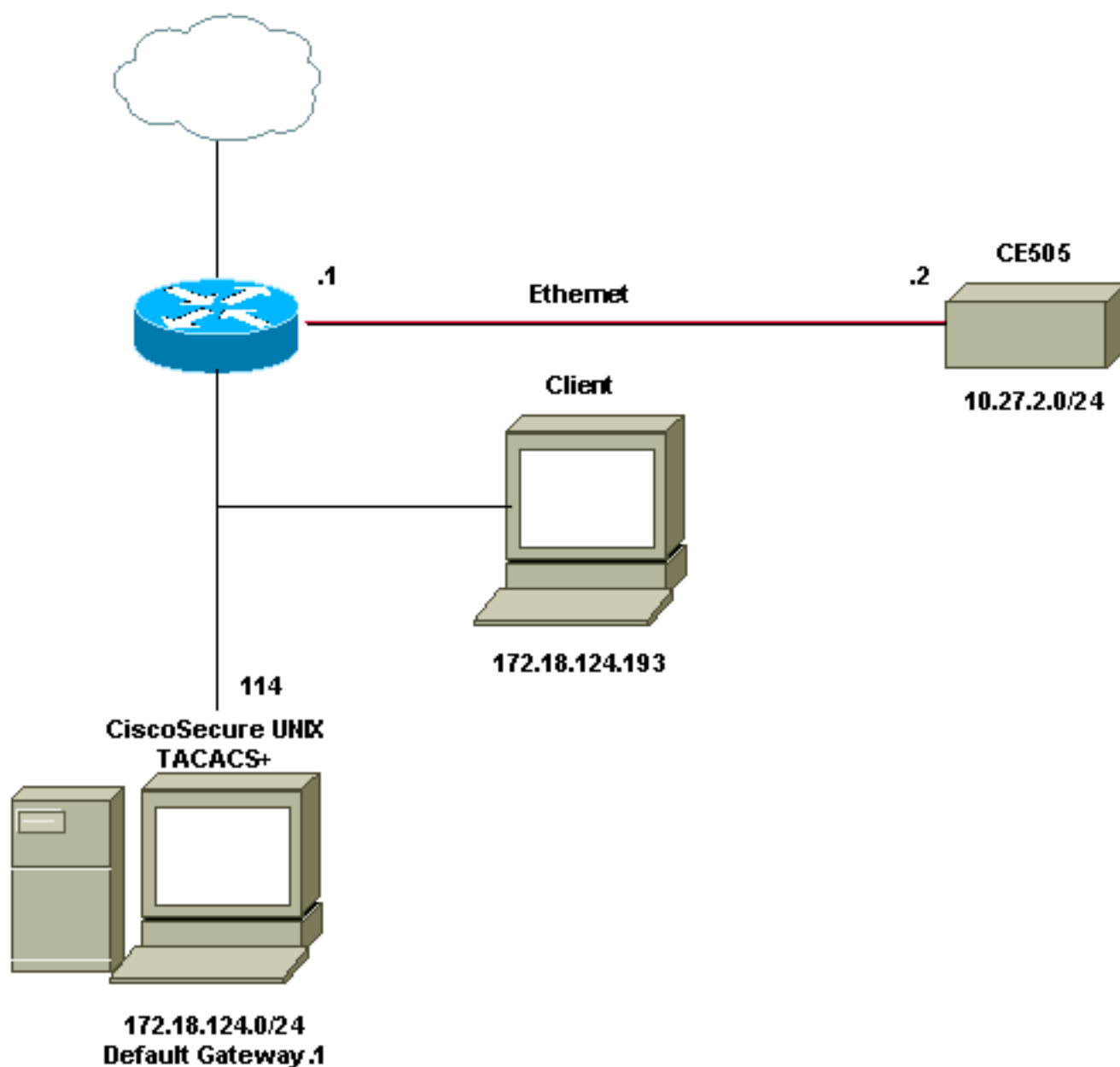
Configurar

En esta sección encontrará la información para configurar las funciones descritas en este documento.

Nota: Utilice la herramienta [Command Lookup Tool](#) ([clientes registrados solamente](#)) para obtener más información sobre los comandos utilizados en esta sección.

Diagrama de la red

En este documento, se utiliza esta configuración de red:



Configure el motor del caché para el soporte de TACACS+

Complete estos pasos para configurar el motor del caché para el soporte de TACACS+:

1. Configuración del Cache Engine para la versión correspondiente del Protocolo de comunicación de la memoria caché de la Web (WCCP).
2. Utilice estos comandos para la configuración predeterminada:

```
authentication login local
enableauthentication configuration local enable
```
3. Configure el dirección IP del servidor TACACS+. Si los servidores múltiples especifican cuál dirección es primaria, entonces los servidores secundarios quedan como opciones en blanco.
4. Configure la autenticación al servidor TACACS+ como primaria. Si el servidor no está disponible, el valor predeterminado será la autenticación especificada localmente.
5. Autenticación de la configuración a la información fundamental TACACS+ en caso necesario.

Nota: Usted debe habilitar el TACACS+ en el Cisco Cache Engine porque los motores del Cisco Cache utilizan el PPP para autenticar con el servidor TACACS, a diferencia del Routers que no requiere el PPP. Para habilitar el TACACS+ en los motores del Cisco Cache, el Cisco Secure ACS abierto 2.6, hacer clic la lengüeta de la **configuración de grupo**, y marcar la casilla de verificación **IP PPP** situada en el área de las configuraciones TACACS+.

Sus líneas de comando deben aparecer similares a esta salida:

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

Verificación

Use esta sección para confirmar que su configuración funciona correctamente.

[La herramienta Output Interpreter Tool \(clientes registrados solamente\)](#) (OIT) soporta ciertos comandos show. Utilice la OIT para ver un análisis del resultado del comando show.

- **¿muestre la versión?** Visualiza el software que se ejecuta en el motor del caché, así como algunos otros componentes como el tiempo de actividad del sistema (tal como donde el código fue iniciado previamente y la fecha en que fue compilado).

```
cepro#show versionCisco
Cache EngineCopyright (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 bbalagotImage text-base 0x108000,
data_base 0x437534System restarted by ReloadThe system has been up for 20 hours, 42 minutes,
59 seconds.System booted from "flash"
```
- **¿muestre el hardware?** Visualiza la misma información que el comando **show version**, así como a los componentes de hardware del motor del caché.

```
cepro#show hardwareCisco Cache
EngineCopyright (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 bbalagotImage text-base 0x108000, data_base
0x437534System restarted by ReloadThe 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)
```
- **¿muestre los ejecutar-config?** Visualiza la configuración corriente en el motor del caché.

```
cepro#show running-configBuilding configuration...Current configuration:!!!user add
admin uid 0 password 1 "eeSdy9dcy" capability admin-accessuser 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 10.27.2.255exit!!interface ethernet
1exit!ip default-gateway 10.27.2.1ip route 0.0.0.0 0.0.0.0 10.27.2.1cron file
/local/etc/crontab!wccp router-list 1 10.27.2.1wccp web-cache router-list-num
1!authentication login tacacs enable primaryauthentication login local enable !--- on by
default ---!authentication configuration tacacs enableauthentication configuration local
```

```
enable !---- on by default ---!tacacs server 172.18.124.114 primaryrule no-cache url-regex
.*cgi-bin.*rule no-cache url-regex .*aw-cgi.*!!endcepro#
```

- **¿muestre los tacacs?** Visualiza las configuraciones para el servidor 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
```

- **¿muestre los tacacs de las estadísticas?** Estadísticas de las visualizaciones


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

- **¿muestre la autenticación?** Visualiza la configuración actual actual de la autenticación y autorización TACACS+. `cepro#show authentication`

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

Comandos para resolución de problemas

En esta sección encontrará información que puede utilizar para solucionar problemas de configuración.

[La herramienta Output Interpreter Tool \(clientes registrados solamente\)](#) (OIT) soporta ciertos comandos show. Utilice la OIT para ver un análisis del resultado del comando show.

Nota: Consulte [Información Importante sobre Comandos de Debug](#) antes de usar un comando debug.

- **¿muestre el debug?** Visualiza los comandos debug se habilitan que. `cepro#show debug`

```
Authentication debugging is onTacacs debugging is on
```
- **¿monitor terminal?** Visualiza las salidas de los debugs a la pantalla. Esta salida visualiza los resultados de los comandos **debug authentication and debug tacacs**. `cepro#terminal monitor`

```
cepro#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() Begintacacs_plus_socket()
Begintacacs_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()
Begintacacs_plus_connect() socket(28) 784tacacs_plus_connect() Endopen_handle() is
connectedopen_handle() *connection_handle 28open_handle() **pSocketHandleIndex
28tacacs_plus_openconn() **pSocketHandleIndex 28get_server() Begintacacs_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() Begintacacs_plus_proc_send_receive()
Begintacacs_plus_proc_send_receive() length 33copy_tac_plus_packet() Begintacacs_malloc()
Begin 45tacacs_malloc() PSkmalloc ptrcopy_tac_plus_packet() malloc_named
copytacacs_plus_encrypt() BegingetTacacsKey(): BegingetTacacsKey(): 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() ntohs(out_pak->length) 33dump_start_session() Begin
ntohs(out_pak->length) 33getTacacsKey(): BegingetTacacsKey(): 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() Beginencrypt_md5_xor() no keydump_summarise_incoming_packet_type()
BeginRead AUTHEN/START size=45dump_nas_pak() Begindump_header() BeginPACKET: 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(): BegingetTacacsTimeout(): ulTimeout = 5tacacs_plus_sockwrite()
Begintacacs_plus_proc_send_receive() PSkfree out_paketTacacsTimeout():
BegingetTacacsTimeout(): ulTimeout = 5sockread() Begintacacs_plus_proc_send_receive()
readtacacs_malloc() Begin 18tacacs_malloc() PSkmalloc ptrtacacs_plus_proc_send_receive()
malloc_named *intacacs_plus_proc_send_receive() allocated memorygetTacacsTimeout():
BegingetTacacsTimeout(): ulTimeout = 5sockread() Begintacacs_plus_proc_send_receive()
OKtacacs_plus_decrypt() BegingetTacacsKey(): BegingetTacacsKey(): sKey
=tacacs_plus_decrypt() keytacacs_plus_decrypt() tac_pak->encrypted = TAC_PLUS_CLEAR && key
is emptyauthen_resp_sanity_check() Begintacacs_plus_hdr_sanity_check()
Beginauthen_debug_response() Beginauthen_debug_response() TACACS+: ver=192 id=1541646967
received AUTHEN      status = FAILtacacs_plus_start_login() PSkfree
out_tac_pakunload_authen_resp() Begintacacs_plus_start_login() PSkfree
in_tac_pakdebug_authen_status() BeginTACACS+/AUTHEN (2007688027): status =
FAILtacacs_plus_login() Authentication failed.tacacs_plus_login() labellaaa_cleanup_login()
Beginaaa_close_connection() Begintacacs_plus_closeconn() Beginget_server()
Beginclose_handle() Beginclose_handle() nHandleIndex 28 nSockFdTbl[**handle]
784aaa_set_password() Beginaaa_free_user() Begindebug_authen_svc()
Beginaaa_close_connection() BeginTACACS+/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() Begindebug_start_login()
Begindebug_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() Begintacacs_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
```

[Información Relacionada](#)

- [Release Note para el Software Cisco Cache, versión 2.3.0 - Memoria caché de FTP](#)
- [Productos y servicios del Cisco 500 Series Cache Engine](#)
- [Soporte Técnico y Documentación - Cisco Systems](#)