PIX 6.2 : Configuratievoorbeeld van verificatie en autorisatie

Inhoud

Inleiding
Voordat u begint
Conventies
Voorwaarden
Gebruikte componenten
Test voordat u verificatie/autorisatie toevoegt
Indelingsinstellingen begrijpen
Verificatie/autorisatie - Lokale gebruikersnamen
Verificatie/autorisatie met een AAA-server
ACS - TACACS+
CSUnix - TACACS+
ACS - RADIUS
CSUnix - RADIUS
Netwerktoegangsbeperkingen
Debuggen
accounting
Te verzamelen informatie als u een TAC-case opent
Gerelateerde informatie

Inleiding

PIX-opdrachttoestemming en uitbreiding van lokale echtheidscontrole zijn ingevoerd in versie 6.2. Dit document geeft een voorbeeld van hoe dit op een PIX moet worden ingesteld. Eerder beschikbare authenticatie functies zijn nog beschikbaar maar niet besproken in dit document (bijvoorbeeld Secure Shell (SSH), IPsec client-verbinding vanaf een pc, enzovoort). De uitgevoerde opdrachten kunnen lokaal op de PIX worden geregeld of op afstand via TACACS+. RADIUS-opdrachtautorisatie wordt niet ondersteund; dit is een beperking van het RADIUS-protocol.

De lokale opdrachtautorisatie wordt uitgevoerd door opdrachten en gebruikers toe te wijzen aan voorkeursniveaus.

De afstandsbediening maakt gebruik van een TACACS+ verificatie, autorisatie en accounting (AAA) server. Meervoudige AAA-servers kunnen worden gedefinieerd in het geval dat ze onbereikbaar zijn.

Verificatie werkt ook met eerder geconfigureerde IPSec- en SSH-verbindingen. Voor SSH-verificatie dient u deze opdracht uit te geven:

```
aaa authentication ssh console <LOCAL | server_tag>
```

Opmerking: Als u een TACACS+ of RADIUS-servergroep voor verificatie gebruikt, kunt u de PIX configureren om de lokale database als **FALLBACK-**methode te gebruiken als de AAA-server niet beschikbaar is.

Bijvoorbeeld

```
pix(config)#aaa authentication ssh console TACACS+ LOCAL
```

U kunt de lokale database ook gebruiken als hoofdmethode voor verificatie (zonder back-up) als u alleen LOCAL invoert.

Geef deze opdracht bijvoorbeeld uit om een gebruikersaccount in de lokale database te definiëren en lokale authenticatie uit te voeren voor een SSH-verbinding:

```
pix(config)#aaa authentication ssh console LOCAL
```

Raadpleeg Hoe u verificatie kunt uitvoeren en het inschakelen van de Cisco Secure PIX-firewall (5.2 tot en met 6.2) voor meer informatie over hoe u AAA-geauthentiseerde toegang tot een PIX-firewall kunt maken die PIX-softwareversie 5.2 tot en met 6.2 kan uitvoeren en voor meer informatie over verificatie, syslogging en toegang tot de server van AAA uitvalt.

Raadpleeg PIX/ASA : Cut-through Proxy voor Network Access met behulp van TACACS+ en RADIUS Server Configuration Voorbeeld voor meer informatie over hoe u AAA-geauthentiseerde (Cut-Through Proxy) toegang tot een PIX-firewall die PIX-softwareversies 6.3 en hoger uitvoert.

Als de configuratie goed is uitgevoerd, mag u niet uit de PIX worden vergrendeld. Als de configuratie niet is opgeslagen, moet het opnieuw opstarten van de PIX worden teruggebracht naar de pre-configuratie status. Als de PIX niet toegankelijk is vanwege een foutieve configuratie, raadpleegt u de Wachtwoordherstel en de AAA-configuratieservice voor PIX.

Voordat u begint

Conventies

Zie de Cisco Technical Tips Convention voor meer informatie over documentconventies.

Voorwaarden

Er zijn geen specifieke voorwaarden van toepassing op dit document.

Gebruikte componenten

De informatie in dit document is gebaseerd op de volgende software- en hardware-versies:

- PIX-softwarerelease 6.2

- Cisco Secure ACS voor Windows versie 3.0 (ACS)
- Cisco Secure ACS voor UNIX (CSUnix) versie 2.3.6

De informatie in dit document is gebaseerd op apparaten in een specifieke laboratoriumomgeving. Alle apparaten die in dit document worden beschreven, hadden een opgeschoonde (standaard)configuratie. Als u in een levend netwerk werkt, zorg er dan voor dat u de potentiële impact van om het even welke opdracht begrijpt alvorens het te gebruiken.

Test voordat u verificatie/autorisatie toevoegt

Zorg ervoor dat u, voordat u de nieuwe 6.2 verificatie/autorisatie-functies gaat uitvoeren, op dit moment toegang tot de PIX kunt verkrijgen met behulp van deze opdrachten:

```
!--- IP range allowed to Telnet to the PIX (values depend on network). telnet 172.18.124.0
255.255.255.0
!--- Telnet password. passwd <password>
!--- Enable password. enable password <password>
```

Indelingsinstellingen begrijpen

De meeste opdrachten in de PIX-modus zijn op niveau 15, hoewel een paar op niveau 0. Gebruik deze opdracht om de huidige instellingen voor alle opdrachten weer te geven:

```
show privilege all
```

De meeste opdrachten staan standaard op niveau 15, zoals in dit voorbeeld:

```
privilege configure level 15 command route
```

Een paar opdrachten bevinden zich op niveau 0, zoals in dit voorbeeld:

```
privilege show level 0 command curpriv
```

De PIX kan in werking treden om modi in te schakelen en te configureren. Sommige opdrachten, zoals **bloggen tonen**, zijn in beide modi beschikbaar. Om privileges op deze opdrachten in te stellen, moet u de modus specificeren waarin de opdracht bestaat, zoals in het voorbeeld wordt weergegeven. De optie andere modus is **ingeschakeld**. U krijgt de `logbestanden als een opdracht beschikbaar in` foutmelding `voor meerdere modi`. Als u de modus niet configureren gebruikt u de opdracht **Modus [schakelt|configureren]**:

```
privilege show level 5 mode configure command logging
```

Deze voorbeelden richten zich op de opdracht **kloktijd**. Gebruik deze opdracht om de huidige instellingen voor de opdracht **kloktijd** te bepalen:

```
show privilege command clock
```

De uitvoer van de opdracht **showprivilege klokkloktijd** toont aan dat de opdracht **kloktijd** in deze drie formaten bestaat:

```
!--- Users at level 15 can use the show clock command.

privilege show level 15 command clock
!--- Users at level 15 can use the clear clock command.

Privilege clear level 15 command clock
!--- Users at level 15 can configure the clock !--- (for example, clock set 12:00:00 Jan 01
2001).

privilege configure level 15 command clock
```

Verificatie/autorisatie - Lokale gebruikersnamen

Alvorens het voorkeursniveau van de **klokopdracht** te veranderen, zou u naar de troostpoort moeten gaan om een administratieve gebruiker te configureren en LOKALE inlogverificatie aan te zetten, zoals in dit voorbeeld wordt getoond:

```
GOSS(config)# username poweruser password poweruser privilege 15
GOSS(config)# aaa-server LOCAL protocol local
GOSS(config)# aaa authentication telnet console LOCAL
```

De PIX bevestigt de toevoeging van de gebruiker, zoals in dit voorbeeld wordt getoond:

```
GOSS(config)# 502101: New user added to local dbase:
      Uname: poweruser Priv: 15 Encpass: Nimjl8wRa7VAmpm5
```

De gebruiker "Power user" moet in staat zijn om net in de PIX te tellen en met de bestaande lokale PIX-instelling het wachtwoord mogelijk te maken (de **licentie** van **wachtwoord <*wachtwoord*>** opdracht).

U kunt meer beveiliging toevoegen door authenticatie toe te voegen voor het inschakelen, zoals in dit voorbeeld wordt getoond:

```
GOSS(config)# aaa authentication enable console LOCAL
```

De gebruiker moet het wachtwoord invoeren voor inloggen en voor het inschakelen. In dit voorbeeld wordt het wachtwoord "poweruser" gebruikt voor zowel de inlognaam als het inschakelen. Gebruiker "poweruser" dient in staat te zijn om in de PIX te tellen en ook met het

lokale PIX-wachtwoord in te schakelen.

Als u wilt dat sommige gebruikers alleen bepaalde opdrachten kunnen gebruiken, moet u een gebruiker met lagere rechten instellen, zoals in dit voorbeeld wordt weergegeven:

```
GOSS(config)# username ordinary password ordinary privilege 9
```

Omdat vrijwel al uw opdrachten standaard op niveau 15 staan, moet u bepaalde opdrachten naar niveau 9 verplaatsen, zodat de 'gewone' gebruikers ze kunnen uitgeven. In dit geval, wil u dat uw niveau 9 gebruiker de opdracht **showklok** kan gebruiken, maar niet de klok aanpast, zoals in dit voorbeeld wordt getoond:

```
GOSS(config)# privilege show level 9 command clock
```

U hebt ook nodig dat uw gebruiker zich kan uitloggen van de PIX (de gebruiker kan op niveau 1 of 9 zijn als u dit wilt doen), zoals in dit voorbeeld wordt getoond:

```
GOSS(config)# privilege configure level 1 command logout
```

U moet de gebruiker kunnen gebruiken om de opdracht **in te schakelen** (de gebruiker is op niveau 1 wanneer u dit probeert), zoals in dit voorbeeld wordt getoond:

```
GOSS(config)# privilege configure level 1 mode enable command enable
```

Door de opdracht **uitschakelen** naar niveau 1 te verplaatsen, kan elke gebruiker tussen niveau 2-15 de schakelmodus verlaten, zoals in dit voorbeeld wordt getoond:

```
GOSS(config)# privilege configure level 1 command disable
```

Als u net zoals de gebruiker "gewoon" wilt tellen en als dezelfde gebruiker instelt (het wachtwoord is ook "gewoon"), moet u de **opdracht** voor **configuratie niveau 1** gebruiken**,** zoals in dit voorbeeld wordt getoond:

```
GOSS# show curpriv
Username : ordinary
Current privilege level : 9
Current Mode/s : P_PRIV
```

Als de oorspronkelijke sessie nog steeds open is (de sessie voorafgaand aan het toevoegen van enige authenticatie) weet de PIX misschien niet wie u bent omdat u aanvankelijk niet inlogde bij een gebruikersnaam. Als dat het geval is, gebruik de opdracht **debug** om berichten over de gebruiker "Enable_15" of "Enable_1" te bekijken als er geen gekoppelde gebruikersnaam is. Daarom is telnet in de PIX als gebruiker "poweruser" (de "level 15" gebruiker) voorafgaand aan het configureren van de opdrachtautorisatie, omdat u zeker moet zijn dat de PIX een gebruikersnaam kan associëren met de opdrachten die worden geprobeerd. U kunt de opdrachtautorisatie testen met deze opdracht:

```
GOSS(config)# aaa authorization command LOCAL
```

De gebruiker 'PowerUser' zou in staat moeten zijn om alle opdrachten in te schakelen, in te schakelen en uit te voeren. De gebruiker "gewoon" moet de opdrachten van de **show kloktijd** kunnen gebruiken, **inschakelen**, **uitschakelen** en **uitloggen**, maar geen andere, zoals in dit voorbeeld wordt getoond:

```
GOSS# show xlate
Command authorization failed
```

Verificatie/autorisatie met een AAA-server

U kunt gebruikers ook echt maken en autoriseren door een AAA-server te gebruiken. TACACS+ werkt het beste omdat de opdracht kan worden goedgekeurd, maar RADIUS kan ook worden gebruikt. Controleer of er eerdere opdrachten van AAA telnet/console in de PIX zijn (voor het geval dat de **LOCAL AAA**-opdracht eerder is gebruikt), zoals in dit voorbeeld:

```
GOSS(config)# show aaa
AAA authentication telnet console LOCAL
AAA authentication enable console LOCAL
AAA authorization command LOCAL
```

Als er vorige AAA telnet/console opdrachten zijn, verwijdert u deze door deze opdrachten te gebruiken:

```
GOSS(config)# no aaa authorization command LOCAL
GOSS(config)# no aaa authentication telnet console LOCAL
GOSS(config)# no aaa authentication enable console LOCAL
```

Zoals bij het configureren van lokale verificatie, test om ervoor te zorgen dat gebruikers in de PIX kunnen tellen door deze opdrachten te gebruiken.

```
telnet 172.18.124.0 255.255.255.0
!--- IP range allowed to telnet to the PIX (values would depend on network). passwd <password>
!--- Telnet password. Enable password <password>
!--- Enable password.
```

Afhankelijk van welke server u gebruikt, moet u de PIX voor authenticatie/autorisatie configureren met een AAA server.

ACS - TACACS+

Configureer ACS om met de PIX te communiceren door de PIX in de netwerkconfiguratie te definiëren met "Verifieer het gebruik" van TACACS+ (voor Cisco IOS® software). De configuratie van de ACS-gebruiker hangt af van de configuratie van de PIX. De ACS-gebruiker moet minimaal zijn ingesteld met een gebruikersnaam en wachtwoord.

Gebruik in de PIX deze opdrachten:

```
GOSS(config)# enable password cisco123
GOSS(config)# aaa-server TACSERVER protocol tacacs+
GOSS(config)# aaa-server TACSERVER (inside) host   timeout 10
GOSS(config)# aaa authentication telnet console TACSERVER
```

Op dit punt zou de ACS-gebruiker in de PIX moeten kunnen tellen, het mogelijk moeten maken met het bestaande wachtwoord op de PIX en alle opdrachten kunnen uitvoeren. Voer de volgende stappen uit:

1. Als er een noodzaak is om PIX in te schakelen voor verificatie met ACS, kiest u **Interface Configuration > Advanced TACACS+ Settings**.
2. Controleer de **geavanceerde TACACS+ functies in het** vakje **Advanced Configuration Opties**.
3. Klik op **Inzenden**. De geavanceerde TACACS+ instellingen zijn nu zichtbaar onder de gebruikersconfiguratie.
4. Stel Max. rechten voor een AAA-client in op niveau 15.
5. Kies het wachtwoord voor de gebruiker inschakelen (wat kan betekenen dat u een afzonderlijk wachtwoord moet configureren).
6. Klik op **Inzenden**.

Gebruik deze opdracht om verificatie via TACACS+ in de PIX in te schakelen:

```
GOSS(config)# aaa authentication enable console TACSERVER
```

Op dit punt zou de ACS-gebruiker in de PIX moeten kunnen tellen en met de machtigingswachtwoord in ACS moeten kunnen worden ingesteld.

Voordat de PIX-opdracht wordt toegevoegd, moet ACS 3.0 worden gepatcheerd. U kunt het patch downloaden van het Software Center (alleen geregistreerde klanten). U kunt ook aanvullende informatie over dit patroon bekijken door toegang te krijgen tot Cisco bug-ID CSCdw78255 (alleen geregistreerde klanten).

Verificatie moet werken voordat een opdrachtvergunning wordt verleend. Als er een opdrachtautorisatie met ACS moet worden uitgevoerd, kiest u **InterfacConfiguration > TACACS+ (Cisco) > Shell (exec) voor gebruiker en/of groep** en klikt u op **Indienen**. De instellingen van de shell opdrachtautorisatie zijn nu zichtbaar onder de gebruiker (of groep) configuratie.

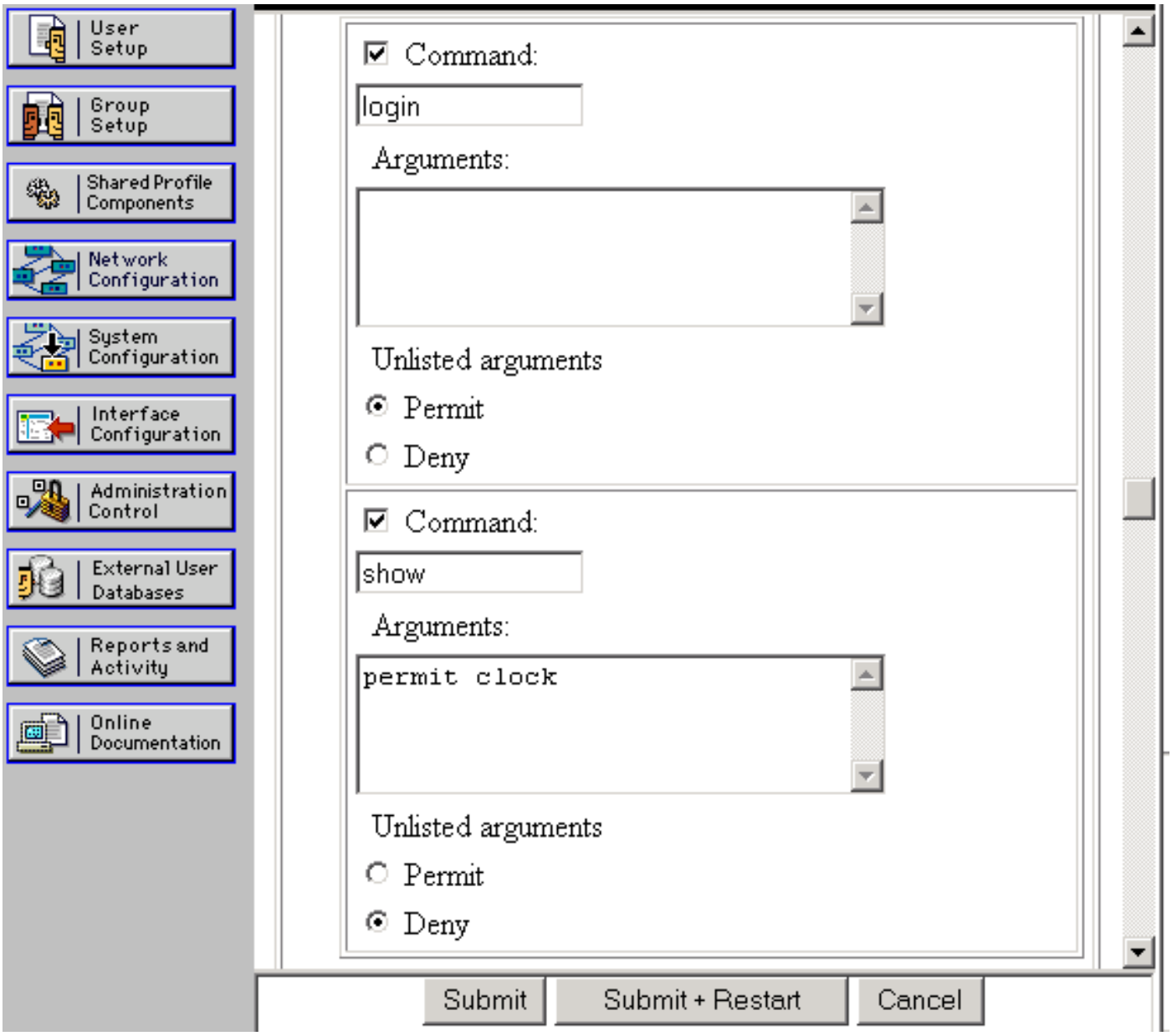
Het is een goed idee om minstens één krachtige ACS-gebruiker in te stellen voor opdrachttoestemming en om niet-afgesloten Cisco IOS-opdrachten toe te staan.

Andere ACS-gebruikers kunnen worden ingesteld met opdrachttoestemming door een subset van opdrachten toe te staan. Dit voorbeeld gebruikt deze stappen:

1. Kies Groepsinstellingen om de gewenste groep uit het uitrolvak te vinden.
2. Klik op **Instellingen bewerken**.
3. Kies **Shell opdrachtautorisatie ingesteld**.
4. Klik op de knop **Opdracht**.
5. Typ **inlognaam**.
6. Kies Toestemming onder niet-vermelde argumenten.
7. Herhaal dit proces voor de **logout**, **stel de** opdrachten in en **schakel** de opties **uit**.
8. Kies Shell commando autorisatie ingesteld.

9. Klik op de knop **Opdracht**.
10. **Entershow**.
11. Voer onder Arguments de **vergunningsklok in**.
12. Kies ontkennen voor niet-vermelde argumenten.
13. Klik op **Inzenden**.

Hier volgen een paar voorbeelden van deze stappen:



Als uw oorspronkelijke sessie nog open is (de sessie voorafgaand aan het toevoegen van enige authenticatie), weet PIX misschien niet wie u bent omdat u aanvankelijk niet inlogde bij een ACS-gebruikersnaam. Als dat het geval is, gebruik de opdracht **debug** om berichten over gebruiker "Enable_15" of "Enable_1" te bekijken als er geen gebruikersnaam is gekoppeld. U dient er zeker van te zijn dat PIX een gebruikersnaam kan associëren met de opdrachten die worden geprobeerd. U kunt dit doen door in de PIX als niveau 15 ACS-gebruiker te telnetteren voordat u een opdrachtautorisatie configureren. U kunt de opdrachtautorisatie testen met deze opdracht:

```
aaa authorization command TACSERVER
```

Op dit punt moet u één gebruiker hebben die in staat moet zijn om alle opdrachten in te

schakelen, in te schakelen en te gebruiken, en een tweede gebruiker die alleen vijf opdrachten kan uitvoeren.

CSUnix - TACACS+

Configureer de CSUnix om met de PIX te communiceren zoals u bij een ander netwerkapparaat zou kunnen doen. De configuratie van de CSUnix-gebruiker is afhankelijk van de configuratie van de PIX. De CSUnix-gebruiker moet minimaal zijn ingesteld met een gebruikersnaam en wachtwoord. In dit voorbeeld zijn drie gebruikers opgericht:

```
!--- This is our "poweruser" who can enable, use all commands, and log in. !--- The login
password is in the 'clear "*********"' statement. !--- The enable password is in the 'clear
"*********" 15' statement. user = pixtest{ password = clear "*********" privilege = clear
"*********" 15 service=shell { default cmd=permit default attribute=permit } } !--- This user can
Telnet in, enable, and use four commands !--- (such as show clock, logout, exit, and enable). !-
-- The login password is in the 'clear "*********"' statement. !--- The enable password is in the
'clear "*********" 15' statement.

user = limitpix{
password = clear "*********"
privilege = clear "*********" 15
service=shell {
cmd=show {
permit "clock"
}
cmd=logout {
permit ".*"
}
cmd=enable {
permit ".*"
}
cmd=exit {
permit ".*"
}
}
}

 !--- This user can Telnet in, but not enable. This user can use any !--- show commands in non-
enable mode as well as logout, exit, and ?.

user = oneuser{
password = clear "*********"
service=shell {
cmd=show {
permit ".*"
}
cmd=logout {
permit ".*"
}
cmd="?" {
permit ".*"
}
cmd=exit {
permit ".*"
}
}
}
```

Gebruik in de PIX deze opdrachten:

```
GOSS(config)# enable password cisco123
GOSS(config)# aaa-server TACSERVER protocol tacacs+
GOSS(config)# aaa-server TACSERVER (inside) host
```

```
GOSS(config)# aaa authentication telnet console TACSERVER
```

Op dit punt moeten alle CSUnix-gebruikers in de PIX kunnen tellen, met het bestaande wachtwoord inschakelen op de PIX en alle opdrachten gebruiken.

Verificatie via TACACS+ in PIX inschakelen:

```
GOSS(config)# aaa authentication enable console TACSERVER
```

Op dit punt zouden de CSUnix-gebruikers die wachtwoorden van "bevoorrecht 15" hebben, in de PIX moeten kunnen tellen en met hen "toelaten" wachtwoorden moeten kunnen inschakelen.

Als uw oorspronkelijke sessie nog steeds open is (de sessie voorafgaand aan het toevoegen van enige authenticatie), weet PIX misschien niet wie u bent omdat u aanvankelijk niet inlogde bij een gebruikersnaam. Als dat het geval is, kan het uitgeven van de **debug** opdracht berichten over gebruiker "Enable_15" of "Enable_1" tonen indien er geen gebruikersnaam is gekoppeld. Telnet in de PIX als gebruiker "elftest" (onze "level 15" gebruiker) voorafgaand aan het configureren van de opdrachtautorisatie, omdat we zeker moeten zijn dat PIX een gebruikersnaam kan associëren met de opdrachten die worden geprobeerd. Verificatie inschakelen moet zijn ingeschakeld voordat u een opdrachtvergunning afgeeft. Als er een opdracht met CSUnix moet worden uitgevoerd, voegt u deze opdracht toe:

```
GOSS(config)# aaa authorization command TACSERVER
```

Van de drie gebruikers kan "elf" alles doen, en de andere twee gebruikers kunnen een subset van opdrachten doen.

ACS - RADIUS

RADIUS-opdrachtautorisatie wordt niet ondersteund. Telnet en laat authenticatie toe is mogelijk met ACS. ACS kan worden ingesteld om met de PIX te communiceren door de PIX in Network Configuration te definiëren met "Authenticate Use" RADIUS (een willekeurige variëteit). De configuratie van de ACS-gebruiker hangt af van de configuratie van de PIX. De ACS-gebruiker moet minimaal zijn ingesteld met een gebruikersnaam en wachtwoord.

Gebruik in de PIX deze opdrachten:

```
GOSS(config)# enable password cisco123
GOSS(config)# aaa-server RADSERVER protocol radius GOSS(config)
            # aaa-server RADSERVER (inside)
              host
```

```
GOSS(config)# aaa authentication telnet console RADSERVER
```

Op dit punt zou de ACS-gebruiker in staat moeten zijn om in de PIX te tellen, om met het bestaande wachtwoord op de PIX in te schakelen en alle opdrachten te gebruiken (de PIX stuurt geen opdrachten naar de RADIUS-server); RADIUS-opdrachtautorisatie wordt niet ondersteund.)

Als u met ACS en RADIUS op de PIX wilt inschakelen, voegt u deze opdracht toe:

```
aaa authentication enable console RADSERVER
```

Anders dan bij TACACS+ wordt hetzelfde wachtwoord gebruikt voor RADIUS-aanmelding zoals bij RADIUS-aanmelding.

CSUnix - RADIUS

Configureer de CSUnix om met de PIX te praten zoals u met een ander netwerkapparaat zou doen. De configuratie van de CSUnix-gebruiker is afhankelijk van de configuratie van de PIX. Dit profiel werkt voor authenticatie en maakt het mogelijk:

```
user = pixradius{
profile_id = 26
profile_cycle = 1
!--- The login password is in the 'clear "********"' statement; !--- this is used for the login,
enable, and non-enable commands.

password = clear "********" < pixradius
}
```

Gebruik in de PIX deze opdrachten:

```
GOSS(config)# enable password cisco123
GOSS(config)# aaa-server RADSERVER protocol radius
GOSS(config)# aaa-server RADSERVER (inside) host
```

Als u ACS en RADIUS in de PIX wilt inschakelen, gebruikt u deze opdracht:

```
GOSS(config)# aaa authentication enable console RADSERVER
```

Anders dan bij TACACS+ wordt hetzelfde wachtwoord gebruikt voor RADIUS-aanmelding zoals bij RADIUS-aanmelding.

Netwerktoegangsbeperkingen

De toegangsbeperkingen van het netwerk kunnen zowel in ACS als in CSUnix worden gebruikt om te beperken wie met de PIX kan verbinden voor administratieve doeleinden.

- **ACS**-De PIX zou worden geconfigureerd in het gebied Netwerktoegangsbeperkingen van de groepsinstellingen. De PIX-configuratie is "Denied Calling/Point of Access Locations" of "Toegelaten Calling/Point of Access Locations" (afhankelijk van het beveiligingsplan).
- **CSUnix** - Dit is een voorbeeld van een gebruiker die toegang tot de PIX is toegestaan, maar geen andere apparaten:

```
user = naruser{
profile_id = 119
profile_cycle = 1
password = clear "********"
privilege = clear "********" 15
service=shell {
allow "10.98.21.50" ".*" ".*"
refuse ".*" ".*" ".*"
default cmd=permit
default attribute=permit
}
}
```

Debuggen

U kunt debug in: gebruik deze opdracht als volgt:

```
logging on
logging
```

Dit zijn voorbeelden van goede en slechte dingen:

- **Good debug** - de gebruiker kan de **inlogoptie** gebruiken, **inschakelen** en **uitvoeren van** opdrachten.
  ```
  307002: Permitted Telnet login session from 172.18.124.111
  111006: Console Login from pixpartial at console
  502103: User priv level changed: Uname: pixpartial From: 1 To: 15
  111009: User 'pixpartial' executed cmd: show clock
  ```
- **Slecht debug**-autorisatie mislukt voor gebruiker, zoals in dit voorbeeld:
  ```
  610101: Authorization failed: Cmd: uauth Cmdtype: show
  ```
- **De externe AAA-server is onbereikbaar:**
  ```
  AAA server host machine not responding
  ```

accounting

Er is geen echte commandoaccounting beschikbaar, maar door syslog geactiveerd te hebben op

de PIX, kunt u zien welke acties zijn uitgevoerd, zoals in dit voorbeeld wordt getoond:

```
307002: Permitted Telnet login session from 172.18.124.111
111006: Console Login from pixtest at console
611103: User logged out: Uname: pixtest
307002: Permitted Telnet login session from 172.18.124.111
111006: Console Login from pixtest at console
502103: User priv level changed: Uname: pixtest From: 1 To: 15
111008: User 'pixtest' executed the 'enable' command.
111007: Begin configuration: 172.18.124.111 reading from terminal
111008: User 'pixtest' executed the 'configure t' command.
111008: User 'pixtest' executed the 'write t' command.
```

Te verzamelen informatie als u een TAC-case opent

Als u nog steeds hulp nodig hebt nadat u de bovenstaande stappen voor het oplossen van problemen hebt gevolgd en u een case wilt openen met Cisco TAC, zorg er dan voor dat u de volgende informatie bevat voor het oplossen van uw PIX-firewall.
• Probleembeschrijving en relevante topologgegevens • Probleemoplossing uitgevoerd voordat u de case opent • Uitvoer vanuit de opdracht **Tech-support** • Uitvoer van het bevel van het **showlogbestand** na het lopen met de **het registreren gebufferde** het bevel, of console vangt die het probleem (indien beschikbaar) aantoont Hang de verzamelde gegevens aan uw case in een niet-zipped, onbewerkte tekstindeling (.txt). U kunt informatie aan uw case toevoegen door deze te uploaden met behulp van de Case Query Tool (alleen geregistreerde klanten). Als u geen toegang hebt tot de Case Query Tool, kunt u de informatie in een e-mailbijlage naar attach@cisco.com met uw casenummer in de onderwerpregel of uw bericht verzenden.

Gerelateerde informatie

- PIX-opdracht
- Cisco PIX-firewall-software - technische ondersteuning en documentatie
- Cisco Secure Access Control Server voor Windows - technische ondersteuning en documentatie
- Cisco Secure Access Control Server voor Unix - technische ondersteuning en documentatie