PIX 6.2: Exemplo do comando Configuration da authentication e autorização

Índice

Introdução

A autorização do comando PIX e a expansão da autenticação local foram introduzidas na versão 6.2. Este documento fornece um exemplo de como configurar isso em um PIX. Os recursos de autenticação disponibilizados anteriormente ainda estão disponíveis, porém, não foram abordados neste documento (Secure Shell (SSH), conexão de cliente IPSec em um PC, etc.) Os comandos executados podem ser localmente controlados no PIX ou remotamente por meio do TACACS+. A autorização de comando RADIUS não é suportada; esta é uma limitação do protocolo RADIUS.

A autorização local dos comandos é feita atribuindo-se comandos e usuários a níveis de privilégio.

A autorização de comando remota é feita através de uma autenticação de TACACS+, autorização e servidor de relatório (AAA). Múltiplos servidores AAA podem ser definidos no caso de um estar inalcançável.

A autenticação também funciona com conexões IPSec e SSH anteriormente configuradas. A autenticação SSH exige que você emite este comando:

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

Nota: Se você usa um TACACS+ ou um grupo de servidor Radius para a autenticação, você pode configurar o PIX para usar o base de dados local como um método da **RESERVA** se o servidor AAA é não disponível.

Por exemplo

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

Você pode alternativamente usar o base de dados local como seu método principal da autenticação (sem a reserva) se você entra em sozinho LOCAL.

Por exemplo, execute este comando para definir uma conta de usuário no banco de dados local e executar a autenticação local para uma conexão de SSH:

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

Refira como executar a autenticação e ativação no firewall PIX segura Cisco (5.2 a 6.2) para obter mais informações sobre de como criar o acesso autenticado AAA a um PIX Firewall que executa a versão de software de PIX 5.2 a 6.2 e para obter mais informações sobre de permita a autenticação, a informações de syslog, e aceder quando o servidor AAA está para baixo.

Refira o PIX/ASA: Corte-através do proxy para o acesso de rede usando o TACACS+ e o exemplo da configuração de servidor RADIUS para obter mais informações sobre de como criar AAA-autenticou (Corte-através do proxy) o acesso a um PIX Firewall que executa as versões de software de PIX 6.3 e mais atrasado.

Se a configuração for realizada corretamente, você não deverá bloquear o PIX. Se a configuração não salvar, recarregar o PIX deve retorná-lo a seu estado da PRE-configuração. Se o PIX estiver inacessível devido a um erro de configuração, consulte Recuperação de Senha e Procedimento de Recuperação de Configuração de AAA para PIX.

Antes de Começar

Convenções

Para obter mais informações sobre convenções de documento, consulte as Convenções de dicas técnicas Cisco.

Pré-requisitos

Não existem requisitos específicos para este documento.

Componentes Utilizados

As informações neste documento são baseadas nestas versões de software e hardware:

- Software PIX versão 6.2
- 3.0 da versão do Cisco Secure ACS for Windows (ACS)
- Cisco Secure ACS para a versão 2.3.6 de UNIX (CSUnix)

As informações neste documento foram criadas a partir de dispositivos em um ambiente de laboratório específico. Todos os dispositivos utilizados neste documento foram iniciados com uma configuração (padrão) inicial. Se você estiver trabalhando em uma rede ativa, certifique-se de que entende o impacto potencial de qualquer comando antes de utilizá-lo.

Testando antes de adicionar a autenticação/autorização

Antes de executar as 6.2 características novas da autenticação/autorização, certifique-se de que você pode atualmente aceder ao PIX usando estes comandos:

```
!--- 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>
```

Entendendo configurações de privilégios

A maioria de comandos no PIX estão a nível 15, embora alguns estejam a nível 0. Para ver configurações atual para comandos all, use este comando:

```
show privilege all
```

A maioria de comandos estão a nível 15 à revelia, segundo as indicações deste exemplo:

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

Alguns comandos estão a nível 0, segundo as indicações deste exemplo:

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

O PIX pode operar-se dentro permite e configura modos. Alguns comandos, tais como a **mostra que registra**, estão disponíveis nos ambos os modos. Para ajustar privilégios nestes comandos, você deve especificar o modo que o comando existe dentro, segundo as indicações do exemplo. A outra opção do modo é **permite**. Você obtém o `registro é um comando disponível no` Mensagem de Erro dos `modos múltiplos`. Se você não configura o modo, use o **modo [permita|configurar] o** comando:

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

Estes exemplos endereçam o **comando clock**. Use este comando determinar as configurações atual para o **comando clock**:

```
show privilege command clock
```

A saída do **comando clock do comando show privilege** mostra que o **comando clock** existe nestes três formatos:

```
!--- 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
```

Autenticação/autorização – Nomes de usuários locais

Antes de mudar o nível de privilégio do **comando clock**, você deve ir à porta de Console configurar um usuário administrativo e girar sobre a autenticação de login local, segundo as indicações deste exemplo:

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

O PIX confirma a adição do usuário, segundo as indicações deste exemplo:

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

O usuário “poweruser” deve poder ao telnet no PIX e para permitir com o PIX local existente permita a senha (essa do **comando password <password> da possibilidade**).

Você pode adicionar mais Segurança adicionando a autenticação para permitir, segundo as indicações deste exemplo:

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

Isto exige o usuário incorporar a senha amba para o início de uma sessão e permiti-la. Neste exemplo, a senha “poweruser” é usada para o início de uma sessão e permite. O usuário "avançado" deve conseguir fazer Telnet no PIX e ainda habilitar a senha local de PIX.

Se você quer alguns usuários poder usar somente determinados comandos, você tem que estabelecer um usuário com mais baixos privilégios, segundo as indicações deste exemplo:

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

Como praticamente todos os comandos estão em um nível 15, por padrão, você deve mover alguns comandos abaixo do nível 9, para que usuários "comuns" possam emiti-los. Nesta instância, você quer seu usuário do nível 9 poder usar o **comando show clock**, mas não reconfigurar o pulso de disparo, segundo as indicações deste exemplo:

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

Você igualmente precisa seu usuário de poder logout do PIX (o usuário pôde estar no nível 1 ou 9 ao querer fazer isto), segundo as indicações deste exemplo:

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

Você precisa o usuário de poder usar o **comando enable** (o usuário está a em nível 1 ao tentar isto), segundo as indicações deste exemplo:

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

Movendo o **comando disable** para o nível 1, todo o usuário entre níveis 2-15 pode sair do modo enable, segundo as indicações deste exemplo:

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

Se você telnet dentro como o usuário “comum barato” e permitir como o mesmo usuário (a senha é igualmente “comum barato”), você usa o **privilégio configura o desabilitação nivelado do comando 1**, segundo as indicações deste exemplo:

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

Se a sessão original ainda estiver aberta (a anterior à autenticação), é possível que o PIX não o reconheça por você não ter feito o login inicialmente com um nome de usuário. Se aquele é o caso, use o **comando debug** ver mensagens sobre o usuário "enable_15" ou "enable_1" se não há nenhum username associado. Portanto, faça um Telnet para o PIX como o usuário "poweruser" (o usuário de "nível 15") antes de configurar a autorização de comando, porque é necessário ter certeza de que o PIX pode associar um nome de usuário aos comandos que estão sendo tentados. Você está pronto à autorização do comando test usando este comando:

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

O usuário "poweruser" deve poder realizar o Telnet, ativar e executar todos os comandos. O

usuário “comum barato” deve poder usar o **pulso de disparo da mostra**, **permite**, **desabilita**, e **comandos logout** mas não outro, segundo as indicações deste exemplo:

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

Autenticação/autorização com um servidor AAA

Também é possível autenticar e autorizar usuários utilizando um servidor AAA. TACACS+ funciona melhor, já a autorização de comando é possível, mas o RADIUS também pode ser usado. Verifique para ver se há Telnet AAA anterior/comandos console no PIX (caso o **comando local aaa** esteve usado previamente), segundo as indicações deste exemplo:

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

Se há Telnet AAA anterior/comandos console, remova-os usando estes comandos:

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

Como com configurar a autenticação local, o teste para certificar-se de usuários enlata o telnet no PIX usando estes comandos.

```
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.
```

Segundo que server você está usando, configurar o PIX para a autenticação/autorização com um servidor AAA.

ACS - TACACS+

Configurar o ACS para comunicar-se com o PIX definindo o PIX na configuração de rede com “autenticam usando” o TACACS+ (para o software de Cisco IOS®). A configuração do usuário ACS depende da configuração do PIX. Pelo menos, o usuário ACS deve estabelecer-se com um nome de usuário e senha.

No PIX, use estes comandos:

```
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
```

Neste momento, o usuário ACS deve poder ao telnet no PIX, para permiti-lo com a existência permita a senha no PIX, e execute comandos all. Conclua estes passos:

1. Se há uma necessidade de fazer o PIX permita a autenticação com ACS, escolhem **Interface Configuration > Advanced Tacacs+ Settings**.
2. Verifique as **características do TACACS+ avançado na** caixa das **opções de configuração avançadas**.
3. Clique em Submit. As configurações avançadas de TACAS+ são agora visíveis sob a configuração do usuário.
4. Ajuste o privilégio máximo para todo o cliente de AAA ao nível 15.
5. Escolha o esquema da senha da possibilidade para o usuário (que poderia envolver configurar um separado permite a senha).
6. Clique em Submit.

Para girar sobre permita a autenticação com o TACACS+ no PIX, usam este comando:

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

Neste momento, o usuário ACS deve poder ao telnet no PIX e permitir com a senha da possibilidade configurada no ACS.

Antes de adicionar a autorização do comando pix, o 3.0 ACS deve ser remendado. Você pode transferir a correção de programa do centro de software (clientes registrados somente). Você pode igualmente ver a informação adicional sobre esta correção de programa pela identificação de bug Cisco de acesso CSCdw78255 (clientes registrados somente).

Énecessário que a autenticação esteja funcionando antes que se faça a autorização de comandos. Se há uma necessidade de executar o comando authorization com o ACS, escolha **Interface Configuration > Tacacs+ (Cisco) > shell (exec) para o usuário e/ou o grupo** e o clique **submetem-se**. Os ajustes da autorização do comando shell são agora visíveis sob a configuração do usuário (ou o grupo).

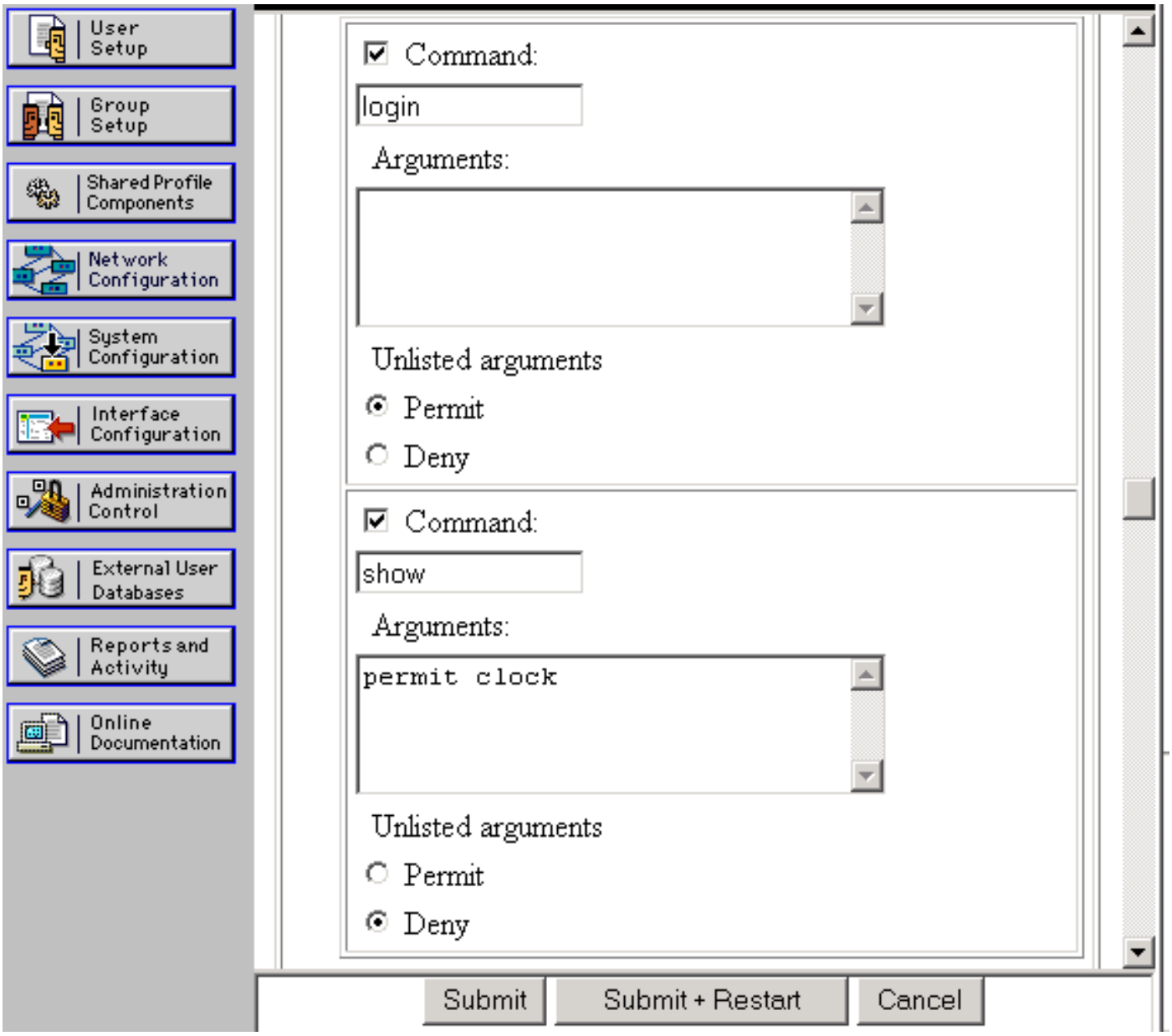
Éuma boa ideia estabelecer pelo menos um usuário poderoso ACS para o comando authorization e permitir comandos cisco ios ímpares.

Outros usuários ACS podem estabelecer-se com comando authorization permitindo um subconjunto dos comandos. Este exemplo usa estas etapas:

1. Escolha configurações de grupo encontrar o grupo desejado da caixa suspensa.
2. O clique **edita ajustes**.
3. Escolha o **grupo da autorização do comando shell**.
4. Clique o **botão comando**.
5. Incorpore o **início de uma sessão**.
6. Escolha a licença sob argumentos não listados.
7. Repita este processo para a **saída**, **permita-o**, e **comandos disable**.
8. Escolha o grupo da autorização do comando shell.
9. Clique o **botão comando**.
10. Entershow.
11. Sob argumentos, entre no **pulso de disparo da licença**.

12. Choose nega para argumentos não listados.
13. Clique em Submit.

Está aqui um exemplo destas etapas:



Se você ainda tem sua sessão original aberta (essa antes de adicionar alguma autenticação), o PIX não pode conhecer quem você é porque você não entrou inicialmente com um nome de usuário de ACS. Se aquele é o caso, use o **comando debug** ver mensagens sobre o usuário "enable_15" ou "enable_1" se não há nenhum username associado. Você precisa para ter certeza o PIX pode associar um username com os comandos que estão sendo tentados. Você pode fazer este por Telnetting no PIX como o usuário do nível 15 ACS antes de configurar o comando authorization. Você está pronto à autorização do comando test usando este comando:

```
aaa authorization command TACSERVER
```

Neste momento, você deve ter um usuário que deve poder ao telnet dentro, permitir, e usar todos os comandos, e um segundo usuário que possa somente fazer cinco comandos.

CSUnix - TACACS+

Configurar CSUnix para comunicar-se com o PIX como você com todo o outro dispositivo de rede. A configuração do usuário CSUnix depende da configuração do PIX. Pelo menos, o usuário de CSUnix deve estabelecer-se com um nome de usuário e senha. Neste exemplo, três usuários estabeleceram-se:

```
!--- 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 ".*"
}
}
}
```

No PIX, use estes comandos:

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

Neste momento, alguns dos usuários de CSUnix devem poder ao telnet no PIX, para permitir com a existência permita a senha no PIX, e use todos os comandos.

Permita a autenticação com o TACACS+ no PIX:

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

Nesse ponto, os usuários do CSUnix que tiverem senhas de "privilégio 15" devem ser capazes de estabelecer uma sessão Telnet no PIX e habilitarem essa sessão com senhas do tipo "enable".

Se a sessão original ainda estiver aberta (a anterior à autenticação), é possível que o PIX não o reconheça por você não ter feito o login inicialmente com um nome de usuário. Se for esse o caso, a emissão do comando debug pode mostrar mensagens sobre user "enable_15" ou "enable_1" caso não haja nome de usuário associado. Efetue Telnet no PIX como usuário "pixtest" (nosso usuário "nível 15") antes de configurar a autorização de comandos, pois precisamos ter certeza de que o PIX pode associar um nome de usuário aos comandos que estão sendo tentados. A habilitação da autenticação deve ser anterior à autorização do comando. Se há uma necessidade de executar o comando authorization com o CSUnix, adicionar este comando:

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

Dos três usuários, “mais pixtest” pode fazer tudo, e outros dois usuários podem fazer um subconjunto dos comandos.

ACS - RADIUS

A autorização do comando radius não é apoiada. O telnet e permite a autenticação é possível com ACS. O ACS pode ser configurado para comunicar-se com o PIX definindo o PIX na configuração de rede com “autentica usando” o RAIO (alguma variedade). A configuração do usuário ACS depende da configuração do PIX. Pelo menos, o usuário ACS deve estabelecer-se com um nome de usuário e senha.

No PIX, use estes comandos:

```
GOSS(config)# enable password cisco123
GOSS(config)# aaa-server RADSERVER protocol radius GOSS(config)
            # aaa-server RADSERVER (inside)
              host <ip> <key> timeout 10
GOSS(config)# aaa authentication telnet console RADSERVER
```

Neste momento, o usuário ACS deve poder ao telnet no PIX, para permitir com a existência permita a senha no PIX, e use comandos all (o PIX não faz comandos send ao servidor Radius; A autorização do comando radius não é apoiada).

Se você quer permitir com ACS e RAIO no PIX, adicionar este comando:

```
aaa authentication enable console RADSERVER
```

Ao contrário com do TACACS+, a mesma senha é usada para o habilitado para radius quanto para ao login radius.

CSUnix - RADIUS

Configurar CSUnix para falar ao PIX como você com todo o outro dispositivo de rede. A configuração do usuário CSUnix depende da configuração do PIX. Este perfil trabalha para a autenticação e ativação:

```
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
}
```

No PIX, use estes comandos:

```
GOSS(config)# enable password cisco123
GOSS(config)# aaa-server RADSERVER protocol radius
GOSS(config)# aaa-server RADSERVER (inside) host <ip> <key> timeout 10
```

Se você quer permitir com ACS e RAIO no PIX, use este comando:

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

Ao contrário com do TACACS+, a mesma senha é usada para o habilitado para radius quanto para ao login radius.

Restrições de acesso à rede

As limitações do acesso de rede podem ser usadas no ACS e no CSUnix para limitar quem pode conectar ao PIX para propósitos administrativos.

- **ACS** — O PIX seria configurado na área de restrições do acesso de rede das configurações de grupo. A configuração de PIX é “Denied Calling/Point of Access Locations” ou “Permitted Calling/Point of Access Locations” (segundo o plano da Segurança).
- **CSUnix** — Este é um exemplo de um usuário que seja acesso permitido ao PIX, mas não de outros dispositivos:
  ```
  GOSS(config)# aaa authentication enable console RADSERVER
  ```

Debug

Para girar sobre debugar, use este comando:

```
logging on
logging <console|monitor> debug
```

Estes são exemplos de bom e de debug ruim:

- **Debug correto** — O usuário pode usar o **início de uma sessão**, **permitir**, e **executar** comandos.

  ```
  logging on
  logging <console|monitor> debug
  ```

- **Debug ruim** — A autorização falha para o usuário, segundo as indicações deste exemplo:

  ```
  logging on
  logging <console|monitor> debug
  ```

- **Não é possível chegar ao servidor remoto AAA:**

  ```
  logging on
  logging <console|monitor> debug
  ```

Relatório

Não há nenhum explicar real do comando disponível, mas tendo o Syslog ativado no PIX, você pode ver que ações foram executadas, segundo as indicações deste exemplo:

```
logging on
logging <console|monitor> debug
```

Informações a serem coletadas se você abrir um caso de TAC

Se você ainda precisa o auxílio após ter seguido os passos de Troubleshooting acima e o quer abrir um caso com o tac Cisco, seja certo incluir a informação seguinte para pesquisar defeitos seu PIX Firewall.
• Descrição do problema e detalhes relevantes de topologia • Troubleshooting executado antes da abertura do caso • Saída do **comando show tech-support** • Saída do comando show log após a execução com o comando de depuração de registro colocado em buffer ou capturas do console que demonstram o problema (se disponível) Anexe os dados coletados para o seu caso em um formato não compactado e texto simples (.txt). Você

pode anexar informações para o seu caso, carregando-o com o uso da Case Query Tool (somente clientes registrados). Se você não pode alcançar a ferramenta do Case Query, você pode enviar a informação em um anexo de Email a attach@cisco.com com seu número de caso na linha de assunto de sua mensagem.

Informações Relacionadas

- Referências de comando PIX
- Software do firewall Cisco PIX - Suporte técnico & documentação
- Cisco Secure Access Control Server para Windows - Suporte técnico & documentação
- Cisco Secure Access Control Server para Unix - Suporte técnico & documentação