Konfigurieren von PIX 5.0.x: TACACS+ und RADIUS

Inhalt

Einführung

Die RADIUS- und TACACS+-Authentifizierung kann für FTP-, Telnet- und HTTP-Verbindungen erfolgen. Die Authentifizierung für andere weniger häufig verwendete TCP-Protokolle kann in der Regel durchgeführt werden.

Die TACACS+-Autorisierung wird unterstützt. Die RADIUS-Autorisierung ist nicht aktiviert. Änderungen bei der PIX 5.0-Authentifizierung, -Autorisierung und -Accounting (AAA) gegenüber der vorherigen Version beinhalten die AAA-Abrechnung für anderen Datenverkehr als HTTP, FTP und Telnet.

Voraussetzungen

Anforderungen

Für dieses Dokument bestehen keine speziellen Anforderungen.

Verwendete Komponenten

Dieses Dokument ist nicht auf bestimmte Software- und Hardwareversionen beschränkt.

Konventionen

Weitere Informationen zu Dokumentkonventionen finden Sie in den Cisco Technical Tips Conventions.

Authentifizierung und Autorisierung

- Die Authentifizierung ist der Benutzer.
- Autorisierung ist das, was der Benutzer tun kann.
- Die Authentifizierung *ist* ohne Autorisierung gültig.
- Die Autorisierung ist ohne Authentifizierung *nicht* gültig.

Nehmen Sie beispielsweise an, Sie haben 100 Benutzer im Netzwerk und nur sechs dieser Benutzer sollen FTP, Telnet oder HTTP außerhalb des Netzwerks ausführen können. Weisen Sie den PIX an, ausgehenden Datenverkehr zu authentifizieren, und geben Sie alle sechs Benutzer-IDs auf dem TACACS+/RADIUS-Sicherheitsserver an. Mit einfacher *Authentifizierung* können diese sechs Benutzer mit Benutzername und Kennwort authentifiziert werden, bevor sie das Dialogfeld verlassen. Die anderen vierundneunzig Benutzer können nicht ausgehen. Das PIX fordert Benutzer zur Eingabe von Benutzername/Kennwort auf und übergibt ihren Benutzernamen und ihr Kennwort an den TACACS+/RADIUS-Sicherheitsserver. Je nach Antwort wird die Verbindung geöffnet oder verweigert. Diese sechs Benutzer können FTP, Telnet oder HTTP verwenden.

Angenommen, *einer* dieser drei Benutzer, "Terry", ist nicht vertrauenswürdig. Sie möchten Terry FTP erlauben, aber nicht HTTP oder Telnet nach außen. Dies bedeutet, dass Sie die *Autorisierung* hinzufügen müssen. Das heißt, *was* Benutzer tun können, zusätzlich zur Authentifizierung *der* Benutzer. Wenn Sie dem PIX eine *Autorisierung* hinzufügen, sendet das PIX zunächst Terrys Benutzernamen und das Kennwort an den Sicherheitsserver und sendet dann eine Autorisierungsanfrage, in der dem Sicherheitsserver mitgeteilt wird, was Terry zu tun *versucht*. Wenn der Server korrekt eingerichtet ist, kann Terry "FTP 1.2.3.4" verwenden, aber es wird ihm die Möglichkeit verweigert, "HTTP" oder "Telnet" überall zu verwenden.

Was der Benutzer mit Authentifizierung/Autorisierung auf

Wenn Sie versuchen, von innen nach außen (oder umgekehrt) mit Authentifizierung/Autorisierung auf:

- **Telnet** - Der Benutzer sieht eine Eingabeaufforderung mit dem Benutzernamen, gefolgt von einer Kennwortanfrage. Wenn die Authentifizierung (und Autorisierung) auf dem PIX/Server erfolgreich ist, wird der Benutzer vom Zielhost nach Benutzername und Kennwort gefragt.
- **FTP** - Der Benutzer sieht eine Eingabeaufforderung für den Benutzernamen. Der Benutzer muss "local_username@remote_username" als Benutzernamen und "local_password@remote_password" als Kennwort eingeben. Der PIX sendet den "local_username" und den "local_password" an den lokalen Sicherheitsserver, und wenn die Authentifizierung (und Autorisierung) auf dem PIX/Server erfolgreich ist, werden der "remote_username" und das "remote_password" darüber hinaus an den FTP-Zielserver übergeben.
- **HTTP** - Ein Fenster, das im Browser angezeigt wird und Benutzername und Kennwort anfordert. Wenn die Authentifizierung (und Autorisierung) erfolgreich ist, erreicht der Benutzer die Ziel-Website darüber hinaus. Beachten Sie, dass **Browser Benutzernamen und**

Kennwörter zwischenspeichern.. Wenn es scheint, dass das PIX eine HTTP-Verbindung synchronisieren sollte, dies aber nicht tut, ist es wahrscheinlich, dass die erneute Authentifizierung tatsächlich mit dem Browser "schießen" den zwischengespeicherten Benutzernamen und das Kennwort an den PIX erfolgt, der diese dann an den Authentifizierungsserver weiterleitet. Dieses Phänomen wird im PIX-Syslog und/oder beim Server-Debuggen angezeigt. Wenn Telnet und FTP normal arbeiten, HTTP-Verbindungen jedoch nicht, dann ist dies der Grund dafür.

Für alle Szenarien verwendete Sicherheitsserver-Konfigurationen

Cisco Secure UNIX TACACS-Serverkonfiguration

Stellen Sie sicher, dass Sie die PIX-IP-Adresse oder den vollqualifizierten Domänennamen und -schlüssel in der CSU.cfg-Datei haben.

```
user = ddunlap {
password = clear "rtp"
default service = permit
}

user = can_only_do_telnet {
password = clear "telnetonly"
service = shell {
cmd = telnet {
permit .*
}
}
}

user = can_only_do_ftp {
password = clear "ftponly"
service = shell {
cmd = ftp {
permit .*
}
}
}

user = httponly {
password = clear "httponly"
service = shell {
cmd = http {
permit .*
}
}
}
```

Cisco Secure UNIX RADIUS-Serverkonfiguration

Verwenden Sie die grafische Benutzeroberfläche (GUI), um die PIX-IP-Adresse und den Schlüssel zur Liste der Netzwerkzugriffsserver (NAS) hinzuzufügen.

```
user=adminuser {
radius=Cisco {
check_items= {
```

```
2="all"
}
reply_attributes= {
6=6
}
}
```

Cisco Secure Windows 2.x RADIUS

Gehen Sie folgendermaßen vor:

1. Rufen Sie ein Kennwort im Abschnitt User Setup GUI (Benutzereinrichtung) ab.
2. Legen Sie im Bereich für die GUI der Gruppeneinrichtung das Attribut 6 (Servicetyp) auf Anmelden oder Verwaltung fest.
3. Fügen Sie die PIX-IP in der NAS-Konfigurations-GUI hinzu.

EasyACS TACACS+

Die EasyACS-Dokumentation beschreibt die Einrichtung.

1. Klicken Sie im Gruppenbereich auf **Shell Exec** (um Exec-Berechtigungen zu gewähren).
2. Um dem PIX die Autorisierung hinzuzufügen, klicken Sie unten in der Gruppeneinrichtung auf **Nicht übereinstimmende IOS-Befehle verweigern**.
3. Wählen Sie für jeden Befehl, den Sie zulassen möchten (z. B. Telnet) **neuen Befehl hinzufügen/bearbeiten aus**.
4. Wenn Sie Telnet für bestimmte Standorte zulassen möchten, geben Sie die IP(s) im Argumentabschnitt im Formular "permit #.#.#.#" ein. Um Telnet allen Standorten zu ermöglichen, klicken Sie auf **Alle nicht aufgeführten Argumente zulassen**.
5. Klicken Sie auf **Bearbeitungsbefehl beenden**.
6. Führen Sie die Schritte 1 bis 5 für jeden der zulässigen Befehle aus (z. B. Telnet, HTTP oder FTP).
7. Fügen Sie die PIX-IP im Abschnitt "GUI der NAS-Konfiguration" hinzu.

Cisco Secure 2.x TACACS+

Der Benutzer erhält ein Kennwort im Abschnitt GUI der Benutzereinrichtung.

1. Klicken Sie im Gruppenbereich auf **Shell Exec** (um Exec-Berechtigungen zu gewähren).
2. Um dem PIX die Autorisierung hinzuzufügen, klicken Sie unten in der Gruppeneinrichtung auf **Nicht übereinstimmende IOS-Befehle verweigern**.
3. Wählen Sie für jeden Befehl, den Sie zulassen möchten (z. B. Telnet) **neuen Befehl hinzufügen/bearbeiten aus**.
4. Wenn Sie Telnet für bestimmte Standorte zulassen möchten, geben Sie IP(s) für zulässige Standorte in das Argumentrechteck ein (z. B. "Zulassen 1.2.3.4"). Um Telnet allen Standorten zu ermöglichen, klicken Sie auf **Alle nicht aufgeführten Argumente zulassen**.
5. Klicken Sie auf **Bearbeitungsbefehl beenden**.
6. Führen Sie die vorherigen Schritte für jeden der zulässigen Befehle aus (z. B. Telnet, HTTP und/oder FTP).
7. Fügen Sie die PIX-IP im Abschnitt "GUI der NAS-Konfiguration" hinzu.

Konfiguration des Livingston RADIUS-Servers

Fügen Sie die PIX-IP-Adresse und den Schlüssel zur Client-Datei hinzu.

```
adminuser Password="all"
User-Service-Type = Shell-User
```

RADIUS-Serverkonfiguration vermerken

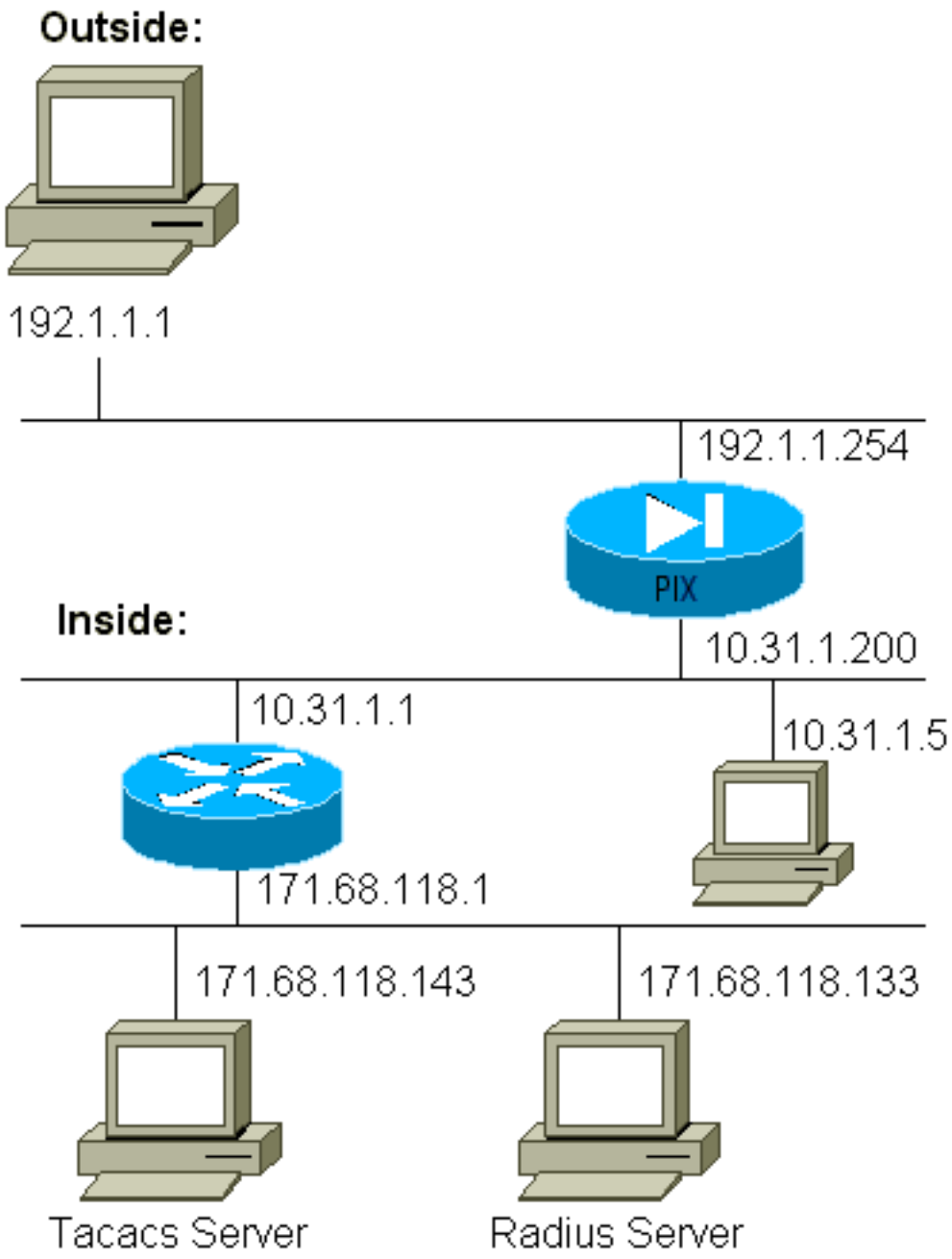
Fügen Sie die PIX-IP-Adresse und den Schlüssel zur Client-Datei hinzu.

```
adminuser Password="all"
Service-Type = Shell-User

key = "cisco"

user = adminuser {
login = cleartext "all"
default service = permit
}

user = can_only_do_telnet {
login = cleartext "telnetonly"
cmd = telnet {
permit .*
}
}

user = httponly {
login = cleartext "httponly"
cmd = http {
permit .*
}
}

user = can_only_do_ftp {
login = cleartext "ftponly"
cmd = ftp {
permit .*
}
}
```

Debugschritte

- Stellen Sie sicher, dass die PIX-Konfigurationen funktionieren, bevor Sie AAA hinzufügen.Wenn Sie keinen Datenverkehr weiterleiten können, bevor Sie Authentifizierung und Autorisierung einsetzen, können Sie dies später nicht mehr tun.
- Aktivieren Sie die Protokollierung in PIX.Der **Debugging-**Befehl **der Protokollierungskonsole** *sollte nicht* auf einem stark ausgelasteten System verwendet werden.Der Befehl **logging buffered debugging** kann verwendet werden. Ausgaben aus den Befehlen **show logging** oder **logging** können an einen Syslog-Server gesendet und geprüft werden.
- Stellen Sie sicher, dass das Debuggen für die TACACS+- oder RADIUS-Server aktiviert ist. Diese Option steht allen Servern zur Verfügung.

Netzwerkdiagramm



PIX-Konfiguration

```
pix-5# write terminal
 nameif ethernet0 outside security0
 nameif ethernet1 inside security100
 enable password 8Ry2YjIyt7RRXU24 encrypted
 passwd 2KFQnbNIdI.2KYOU encrypted
 hostname pixfirewall
 fixup protocol ftp 21
 fixup protocol http 80
 fixup protocol smtp 25
 fixup protocol h323 1720
 fixup protocol rsh 514
 fixup protocol sqlnet 1521
 names
 name 1.1.1.1 abcd
 name 1.1.1.2 a123456789
```

```
 name 1.1.1.3 a123456789123456
 pager lines 24
 logging timestamp
 no logging standby
 logging console debugging
 no logging monitor
 logging buffered debugging
 no logging trap
 logging facility 20
 logging queue 512
 interface ethernet0 auto
 interface ethernet1 auto
 mtu outside 1500
 mtu inside 1500
 ip address outside 192.1.1.254 255.255.255.0
 ip address inside 10.31.1.200 255.255.255.0
 no failover
 failover timeout 0:00:00
 failover ip address outside 0.0.0.0
 failover ip address inside 0.0.0.0
 arp timeout 14400
 global (outside) 1 192.1.1.10-192.1.1.20 netmask
255.255.255.0
 static (inside,outside) 192.1.1.25 171.68.118.143
netmask 255.255.255.255 0 0
 static (inside,outside) 192.1.1.30 10.31.1.5 netmask
255.255.255.255 0 0
 conduit permit tcp any any
 conduit permit icmp any any
 conduit permit udp any any
 no rip outside passive
 no rip outside default
 no rip inside passive
 no rip inside default
 route inside 171.68.118.0 255.255.255.0 10.31.1.1 1
 timeout xlate 3:00:00 conn 1:00:00 half-closed 0:10:00
udp 0:02:00
 timeout rpc 0:10:00 h323 0:05:00
 timeout uauth 0:00:00 absolute
 aaa-server TACACS+ protocol tacacs+
 aaa-server RADIUS protocol radius
 aaa-server AuthInbound protocol tacacs+
 aaa-server AuthInbound (inside) host 171.68.118.143
cisco timeout 5
 aaa-server AuthOutbound protocol radius
 aaa-server AuthOutbound (inside) host 171.68.118.133
cisco timeout 5
 aaa authentication telnet outbound 0.0.0.0 0.0.0.0
0.0.0.0 0.0.0.0 AuthOutbound
 aaa authentication telnet inbound 0.0.0.0 0.0.0.0
0.0.0.0 0.0.0.0 AuthInbound
 aaa authentication http outbound 0.0.0.0 0.0.0.0
0.0.0.0 0.0.0.0 AuthOutbound
 aaa authentication http inbound 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 AuthInbound
 aaa authentication ftp outbound 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 AuthOutbound
 aaa authentication ftp inbound 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 AuthInbound
 no snmp-server location
 no snmp-server contact
 snmp-server community public
 no snmp-server enable traps
 telnet timeout 5
```

```
terminal width 80
Cryptochecksum:fef4bfc9801d7692dce0cf227fe7859b
: end
```

Debug-Beispiele für die Authentifizierung aus PIXAuthentication Debug-Beispielen aus PIX

In diesen Debugbeispielen:

Ausgehend

Der interne Benutzer mit der Adresse 10.31.1.5 initiiert Datenverkehr nach außerhalb von 192.1.1.1 und wird über TACACS+ authentifiziert. Der ausgehende Datenverkehr verwendet die Serverliste "AuthOutbound", die den RADIUS-Server 171.68.118.133 enthält.

Eingehend

Der externe Benutzer unter 192.1.1.1 initiiert den Datenverkehr nach 10.31.1.5 (192.1.1.30) und wird über TACACS authentifiziert. Für eingehenden Datenverkehr wird die Serverliste "AuthInbound" verwendet, die den TACACS-Server 171.68.118.143 umfasst.

PIX Debug - Gute Authentifizierung - TACACS+

Dieses Beispiel zeigt ein PIX-Debugging mit guter Authentifizierung:

```
pixfirewall# 109001: Auth start for user "???" from 192.1.1.1/13155
    to 10.31.1.5/23
 109011: Authen Session Start: user 'pixuser', sid 6
 109005: Authentication succeeded for user 'pixuser' from 10.31.1.5/23
    to 192.1.1.1/13155
 109012: Authen Session End: user 'pixuser', Sid 6, elapsed 1 seconds
 302001: Built inbound TCP connection 6 for faddr 192.1.1.1/13155
    gaddr 192.1.1.30/23 laddr 10.31.1.5/23 (pixuser)
```

PIX Debug - Schlechte Authentifizierung (Benutzername oder Kennwort) - TACACS+

In diesem Beispiel wird das PIX-Debuggen mit fehlerhafter Authentifizierung (Benutzername oder Kennwort) veranschaulicht. Der Benutzer sieht vier Benutzername/Kennwort-Sets und die Meldung "`Fehler: maximale Anzahl von Versuchen überschritten`."

```
pixfirewall# 109001: Auth start for user '???' from 192.1.1.1/13157
    to 10.31.1.5/23
 109006: Authentication failed for user '' from 10.31.1.5/23
    to 192.1.1.1/13157
```

PIX-Debuggen - Ping-Server kann gesendet werden, keine Antwort - TACACS+

In diesem Beispiel wird das PIX-Debuggen veranschaulicht, bei dem der Server gepingt werden kann, jedoch nicht mit dem PIX-Protokoll kommuniziert. Der Benutzer sieht den Benutzernamen einmal, aber PIX fragt nie nach einem Kennwort (dies ist auf Telnet). Der Benutzer sieht "`Fehler:`

Max. Anzahl von Versuchen überschritten."

```
Auth start for user '???' from 192.1.1.1/13159 to
    10.31.1.5/23
 pixfirewall# 109002: Auth from 10.31.1.5/23 to 192.1.1.1/13159
    failed (server 171.68.118.143 failed)
 109002: Auth from 10.31.1.5/23 to 192.1.1.1/13159 failed
    (server 171.68.118.143 failed)
 109002: Auth from 10.31.1.5/23 to 192.1.1.1/13159 failed
    (server 171.68.118.143 failed)
 109006: Authentication failed for user '' from 10.31.1.5/23
    to 192.1.1.1/13159
```

PIX Debug - Ping-Server nicht möglich - TACACS+

Dieses Beispiel zeigt ein PIX-Debuggen, bei dem der Server nicht pingbar ist. Der Benutzer sieht den Benutzernamen einmal, aber der PIX fragt nie nach einem Passwort (dies ist auf Telnet). Diese Meldungen werden angezeigt: "Timeout auf TACACS+-Server" und "Fehler: Max. Anzahl von Versuchen überschritten" (in der Konfiguration haben wir in einem Scheinserver ausgetauscht).

```
109001: Auth start for user '???' from 192.1.1.1/13158
    to 10.31.1.5/23
 109002: Auth from 10.31.1.5/23 to 192.1.1.1/13158 failed
    (server 171.68.118.143 failed)
 109002: Auth from 10.31.1.5/23 to 192.1.1.1/13158 failed
    (server 171.68.118.143 failed)
 109002: Auth from 10.31.1.5/23 to 192.1.1.1/13158 failed
    (server 171.68.118.143 failed)
 109006: Authentication failed for user '' from 10.31.1.5/23
    to 192.1.1.1/13158
```

PIX Debug - Gute Authentifizierung - RADIUS

Dieses Beispiel zeigt ein PIX-Debugging mit guter Authentifizierung:

```
109001: Auth start for user '???' from 10.31.1.5/11074
    to 192.1.1.1/23
 109011: Authen Session Start: user 'pixuser', Sid 7
 109005: Authentication succeeded for user 'pixuser'
    from 10.31.1.5/11074 to 192.1.1.1/23
 109012: Authen Session End: user 'pixuser', Sid 7,
    elapsed 1 seconds
 302001: Built outbound TCP connection 7 for faddr 192.1.1.1/23
    gaddr 192.1.1.30/11074 laddr 10.31.1.5/11074 (pixuser)
```

PIX Debug - Schlechte Authentifizierung (Benutzername oder Kennwort) - RADIUS

Dieses Beispiel zeigt ein PIX-Debuggen mit schlechter Authentifizierung (Benutzername oder Kennwort). Der Benutzer erhält eine Anfrage für Benutzername und Kennwort. Der Benutzer hat drei Möglichkeiten für die erfolgreiche Eingabe von Benutzername/Kennwort.

```
- 'Error: max number of tries exceeded'
 pixfirewall# 109001: Auth start for user '???' from
    192.1.1.1/13157 to 10.31.1.5/23
 109001: Auth start for user '???' from 10.31.1.5/11075
    to 192.1.1.1/23
```

```
 109002: Auth from 10.31.1.5/11075 to 192.1.1.1/23 failed
    (server 171.68.118.133 failed)
 109002: Auth from 10.31.1.5/11075 to 192.1.1.1/23 failed
    (server 171.68.118.133 failed)
 109002: Auth from 10.31.1.5/11075 to 192.1.1.1/23 failed
    (server 171.68.118.133 failed)
 109006: Authentication failed for user '' from 10.31.1.5/11075
    to 192.1.1.1/23
```

Ping Debug - Can Ping Server, Daemon Down - RADIUS

Dieses Beispiel zeigt ein PIX-Debuggen, bei dem der Server pingfähig ist, der Daemon jedoch nicht verfügbar ist und nicht mit dem PIX kommuniziert. Der Benutzer sieht Benutzername, Kennwort und die Meldungen "`RADIUS-Server fehlgeschlagen`" und "`Fehler: Max. Anzahl von Versuchen überschritten.`"

```
pixfirewall# 109001: Auth start for user '???'
    from 10.31.1.5/11076 to 192.1.1.1/23
 109002: Auth from 10.31.1.5/11076 to 192.1.1.1/23 failed
    (server 171.68.118.133 failed)
 109002: Auth from 10.31.1.5/11076 to 192.1.1.1/23 failed
    (server 171.68.118.133 failed)
 109002: Auth from 10.31.1.5/11076 to 192.1.1.1/23 failed
    (server 171.68.118.133 failed)
 109006: Authentication failed for user '' from 10.31.1.5/11076
    to 192.1.1.1/23
```

PIX Debug - Nicht in der Lage, Server oder Schlüssel-/Client-Nichtübereinstimmung zu pingen - RADIUS

In diesem Beispiel wird ein PIX-Debuggen angezeigt, bei dem der Server nicht pingbar ist oder eine Schlüssel-Client-Diskrepanz vorliegt. Der Benutzer sieht Benutzername, Kennwort und die Meldungen "`Timeout to RADIUS server`" und "`Fehler: Max. Anzahl von Versuchen überschritten`" (ein Scheinserver wurde in der Konfiguration ausgetauscht).

```
109001: Auth start for user '???' from 10.31.1.5/11077
    to 192.1.1.1/23
 109002: Auth from 10.31.1.5/11077 to 192.1.1.1/23 failed
    (server 100.100.100.100 failed)
 109002: Auth from 10.31.1.5/11077 to 192.1.1.1/23 failed
    (server 100.100.100.100 failed)
 109002: Auth from 10.31.1.5/11077 to 192.1.1.1/23 failed
    (server 100.100.100.100 failed)
 109006: Authentication failed for user '' from 10.31.1.5/11077
    to 192.1.1.1/23
```

Autorisierung hinzufügen

Wenn Sie die Autorisierung hinzufügen möchten, benötigen Sie die Autorisierung für denselben Quell- und Zielbereich (da die Autorisierung ohne Authentifizierung nicht gültig ist):

```
aaa authorization telnet inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
aaa authorization HTTP inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
aaa authorization ftp inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
```

Beachten Sie, dass die Autorisierung nicht für "ausgehende" Zugriffe hinzugefügt wird, da

ausgehender Datenverkehr mit RADIUS authentifiziert wird und die RADIUS-Autorisierung ungültig ist.

Debug-Beispiele für Authentifizierung und Autorisierung aus PIX

PIX Debug - Gute Authentifizierung und erfolgreiche Autorisierung - TACACS+

Dieses Beispiel zeigt ein PIX-Debugging mit guter Authentifizierung und erfolgreicher Autorisierung:

```
109011: Authen Session Start: user 'pixuser', Sid 8
 109007: Authorization permitted for user 'pixuser'
    from 192.1.1.1/13160 to 10.31.1.5/23
 109012: Authen Session End: user 'pixuser', Sid 8,
    elapsed 1 seconds
 302001: Built inbound TCP connection 8 for faddr 192.1.1.1/13160
    gaddr 192.1.1.30/23 laddr 10.31.1.5/23 (pixuser)
```

PIX Debug - Gute Authentifizierung, fehlgeschlagene Autorisierung - TACACS+

Dieses Beispiel zeigt ein PIX-Debuggen mit guter Authentifizierung, aber mit fehlgeschlagener Autorisierung. Hier sieht der Benutzer auch die Meldung "`Fehler: Autorisierung verweigert.`"

```
109001: Auth start for user '???' from 192.1.1.1/13162
    to 10.31.1.5/23
 109011: Authen Session Start: user 'userhttp', Sid 10
 109005: Authentication succeeded for user 'userhttp'
    from 10.31.1.5/23 to 192.1.1.1/13162
 109008: Authorization denied for user 'userhttp'
    from 10.31.1.5/23 to 192.1.1.1/13162
 109012: Authen Session End: user 'userhttp', Sid 10,
    elapsed 1 seconds
 302010: 0 in use, 2 most used
```

Accounting hinzufügen

TACACS+

```
aaa accounting any inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
```

Debug sieht gleich aus, ob die Accounting aktiviert oder deaktiviert ist. Zum Zeitpunkt des Built-Projekts wird jedoch ein "Start"-Accounting-Datensatz gesendet. Zum Zeitpunkt der "Entfernung" wird ein "Stopp"-Buchungsdatensatz gesendet.

Die TACACS+-Accounting-Datensätze sehen wie folgt aus (sie stammen von Cisco Secure NT, daher das durch Kommata getrennte Format):

```
04/26/2000,01:31:22,pixuser,Default Group,192.1.1.1,
  start,,,,,,,0x2a,,PIX,10.31.1.200,telnet,6,
  Login,1,,,1,,,,,,,,,,,,local_ip=10.31.1.5 foreign_ip=192.1.1.1,
```

```
  ,, ,,,,,,,,,zekie,,,,,,,,^
04/26/2000,01:31:26,pixuser,Default Group,192.1.1.1,stop,4,
  ,36,82,,,0x2a,,PIX,10.31.1.200,telnet,6,
  Login,1,,,1,,,,,,,,,,,local_ip=10.31.1.5 foreign_ip=192.1.1. 1,
  ,,,,,,,,,,,zekie,,,,,,,,
```

RADIUS

```
aaa accounting any outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthOutbound
```

Debug sieht gleich aus, ob die Accounting aktiviert oder deaktiviert ist. Zum Zeitpunkt des Built-Projekts wird jedoch ein "Start"-Accounting-Datensatz gesendet. Zum Zeitpunkt der "Entfernung" wird ein "Stopp"-Buchungsdatensatz gesendet.

RADIUS-Accounting-Datensätze sehen wie diese aus (diese stammen von Cisco Secure UNIX; in Cisco Secure NT können stattdessen durch Kommas getrennt werden):

```
radrecv: Request from host a1f01c8 code=4, id=18, length=65
Acct-Status-Type = Start
Client-Id = 10.31.1.200
Login-Host = 10.31.1.5
Login-TCP-Port = 23
Acct-Session-Id = "0x0000002f"
User-Name = "pixuser"
Sending Accounting Ack of id 18 to a1f01c8 (10.31.1.200)
radrecv: Request from host a1f01c8 code=4, id=19, length=83
Acct-Status-Type = Stop
Client-Id = 10.31.1.200
Login-Host = 10.31.1.5
Login-TCP-Port = 23
Acct-Session-Id = "0x0000002f"
Username = "pixuser"
Acct-Session-Time = 7
```

Verwendung des Befehls Except

Wenn in unserem Netzwerk entschieden wird, dass eine bestimmte Quelle und/oder ein bestimmtes Ziel keine Authentifizierung, Autorisierung oder Abrechnung benötigt, können wir so etwas wie diese Ausgabe ausführen:

```
aaa authentication except inbound 192.1.1.1 255.255.255.255
   0.0.0.0 0.0.0.0 AuthInbound
```

Wenn Sie ein Feld von der Authentifizierung ausnehmen und über eine Autorisierung verfügen, müssen Sie auch das Feld von der Autorisierung ausschließen.

Max. Sitzungen und Anzeigen angemeldeter Benutzer

Einige TACACS+- und RADIUS-Server verfügen über die Funktionen "max-session" (max-session) oder "view login users" (Anzeige angemeldeter Benutzer). Die Möglichkeit, maximal Sitzungen durchzuführen oder angemeldete Benutzer zu überprüfen, hängt von den Accounting-

Datensätzen ab. Wenn ein verbuchter "Start"-Datensatz generiert wird, aber kein "Stopp"-Datensatz vorhanden ist, geht der TACACS+- oder RADIUS-Server davon aus, dass die Person noch angemeldet ist (eine Sitzung über den PIX).

Dies funktioniert aufgrund der Art der Verbindungen gut für Telnet- und FTP-Verbindungen. Dies funktioniert bei HTTP aufgrund der Art der Verbindung nicht gut. In dieser Beispielausgabe wird eine andere Netzwerkkonfiguration verwendet, die Konzepte sind jedoch identisch.

Der Benutzer Telnet über den PIX authentifiziert sich auf dem Weg:

```
(pix) 109001: Auth start for user '???' from 171.68.118.100/1200
    to 9.9.9.25 /23
(pix) 109011: Authen Session Start: user 'cse', Sid 3
(pix) 109005: Authentication succeeded for user 'cse'
    from 171.68.118.100/12 00 to 9.9.9.25/23
(pix) 302001: Built TCP connection 5 for faddr 9.9.9.25/23
    gaddr 9.9.9.10/12 00 laddr 171.68.118.100/1200 (cse)
(server start account) Sun Nov 8 16:31:10 1998
    rtp-pinecone.rtp.cisco.com cse
 PIX 171.68.118.100 start task_id=0x3 foreign_ip=9.9.9.25
    local_ip=171.68.118.100 cmd=telnet
```

Da der Server einen "Start"-Datensatz, aber keinen "Stopp"-Datensatz gesehen hat (zu diesem Zeitpunkt), zeigt der Server an, dass der "Telnet"-Benutzer angemeldet ist. Wenn der Benutzer eine andere Verbindung versucht, die eine Authentifizierung erfordert (möglicherweise von einem anderen PC aus) und für diesen Benutzer auf dem Server auf "1" gesetzt ist (vorausgesetzt, der Server unterstützt max-sessions), wird die Verbindung vom Server abgelehnt.

Der Benutzer fährt mit dem Telnet- oder FTP-Geschäft auf dem Ziel-Host fort und verlässt das System anschließend (verbringt dort 10 Minuten):

```
(pix) 302002: Teardown TCP connection 5 faddr
    9.9.9.25/80 gaddr 9.9.9.10/128 1
    laddr 171.68.118.100/1281 duration 0:00:00 bytes 1907 (cse)
   (server stop account) Sun Nov 8 16:41:17 1998
    rtp-pinecone.rtp.cisco.com cse
 PIX 171.68.118.100 stop task_id=0x3 foreign_ip=9.9.9.25
    local_ip=171.68.118.100 cmd=telnet elapsed_time=5
    bytes_in=98 bytes_out=36
```

Ob uauth 0 (jedes Mal authentifizieren) oder mehr (einmal und nicht einmal während des Wartezeitraums authentifizieren), für jede Website, auf die zugegriffen wird, wird ein Buchhaltungsdatensatz abgeschnitten.

HTTP funktioniert aufgrund der Art des Protokolls anders. Diese Ausgabe zeigt ein Beispiel für HTTP:

Der Benutzer wählt zwischen 171.68.118.100 und 9.9.9.25 mithilfe des PIX:

```
(pix) 109001: Auth start for user '???' from 171.68.118.100/1281
    to 9.9.9.25 /80
 (pix) 109011: Authen Session Start: user 'cse', Sid 5
 (pix) 109005: Authentication succeeded for user 'cse'
    from 171.68.118.100/12 81 to 9.9.9.25/80
 (pix) 302001: Built TCP connection 5 for faddr 9.9.9.25/80
    gaddr 9.9.9.10/12 81 laddr 171.68.118.100/1281 (cse)
```

```
(server start account) Sun Nov 8 16:35:34 1998
   rtp-pinecone.rtp.cisco.com cse
PIX 171.68.118.100 start task_id=0x9 foreign_ip=9.9.9.25
   local_ip=171.68.118.100 cmd=http
(pix) 302002: Teardown TCP connection 5 faddr 9.9.9.25/80
   gaddr 9.9.9.10/128 1 laddr 171.68.118.100/1281 duration
   0:00:00 bytes 1907 (cse)
(server stop account) Sun Nov 8 16:35.35 1998
   rtp-pinecone.rtp.cisco .com cse PIX 171.68.118.100
   stop task_id=0x9 foreign_ip =9.9.9.25
local_ip=171.68.118.100 cmd=http elapsed_time=0
   bytes_ in=1907 bytes_out=223
```

Der Benutzer liest die heruntergeladene Webseite.

Der Startrekord wurde um 16:35:34 veröffentlicht, und der Stoppdatensatz um 16:35:35. Dieser Download dauerte eine Sekunde (d.h. es gab weniger als eine Sekunde zwischen dem Start und dem Stopp Record). Ist der Benutzer immer noch bei der Website angemeldet, und die Verbindung bleibt noch offen, wenn er die Webseite liest? Nein. Funktionieren hier die max. Sitzungen oder die Ansicht der angemeldeten Benutzer? Nein, weil die Verbindungszeit (die Zeit zwischen "Built" und "Teardown") in HTTP zu kurz ist. Der Startdatensatz und der Stopp-Datensatz sind Sekundenbruchteile. Es wird keinen "Start"-Datensatz ohne "Stopp"-Datensatz geben, da die Datensätze praktisch im selben Augenblick auftreten. Es wird immer noch "start" und "stop" Datensatz an den Server für jede Transaktion gesendet, unabhängig davon, ob uauth auf 0 oder etwas größer gesetzt ist. Allerdings funktionieren maximale Sitzungen und die Ansicht angemeldeter Benutzer aufgrund der Art der HTTP-Verbindungen nicht.

Authentifizierung und Aktivierung auf dem PIX selbst

Im vorherigen Abschnitt wurde die Authentifizierung von Telnet- (und HTTP-, FTP-)Datenverkehr *über* den PIX beschrieben. Wir stellen sicher, dass Telnet *zu* PIX *ohne* Authentifizierung funktioniert in:

```
telnet 10.31.1.5 255.255.255.255 passwd ww
```

```
aaa authentication telnet console AuthInbound
```

Wenn Benutzer Telnet an den PIX anschließen, werden sie zur Eingabe des Telnet-Kennworts aufgefordert (**ww**). Anschließend fordert das PIX auch den TACACS+ (in diesem Fall, da die Serverliste "AuthInbound" verwendet wird) oder den RADIUS-Benutzernamen und das RADIUS-Kennwort an. Wenn der Server ausgefallen ist, können Sie in das PIX-System gelangen, indem Sie **pix** für den Benutzernamen eingeben und dann das enable-Kennwort (**enable password *any*) eingeben, um** Zugriff zu erhalten.

Mit diesem Befehl:

```
aaa authentication enable console AuthInbound
```

Der Benutzer wird aufgefordert, einen Benutzernamen und ein Kennwort einzugeben, die an das

TACACS-System gesendet werden (in diesem Fall, da die Serverliste "AuthInbound" verwendet wird, wird die Anforderung an den TACACS-Server gesendet) oder an den RADIUS-Server. Da das Authentifizierungspaket für "enable" mit dem Authentifizierungspaket für die Anmeldung übereinstimmt, kann der Benutzer, wenn er sich mit TACACS oder RADIUS beim PIX anmelden kann, über TACACS oder RADIUS mit demselben Benutzernamen/Kennwort aktivieren. Dieses Problem wurde der Cisco Bug-ID CSCdm47044 zugewiesen (nur registrierte Kunden).

Authentifizierung auf der seriellen Konsole

Der Befehl **aaa authentication serial console AuthInbound** erfordert eine Authentifizierungsüberprüfung, um auf die serielle Konsole des PIX zugreifen zu können.

Wenn der Benutzer Konfigurationsbefehle über die Konsole ausführt, werden Syslog-Meldungen unterbrochen (vorausgesetzt, das PIX ist so konfiguriert, dass Syslog auf Debugebene an einen Syslog-Host gesendet wird). Dies ist ein Beispiel für die Anzeige auf dem Syslog-Server:

```
logmsg: pri 245, flags 0, from [10.31.1.200.2.2], msg Nov 01 1999
 03:21:14: %PIX-5-111008: User 'pixuser' executed the 'logging' command.
```

Ändern der angezeigten Aufforderung für Benutzer

Wenn Sie den Befehl **auth-prompt PIX_PIX_PIX** haben, sehen Benutzer, die den PIX durchlaufen, die folgende Sequenz:

```
PIX_PIX_PIX [at which point one would enter the username]
Password:[at which point one would enter the password]
```

Bei der Ankunft im endgültigen Zielfeld wird die Eingabeaufforderung "Benutzername:" und "Passwort:" angezeigt. Diese Eingabeaufforderung betrifft nur Benutzer, die *durch* den PIX gehen, nicht *auf* den PIX.

Hinweis: Für den Zugriff auf das PIX gibt es keine getrennte Buchhaltung.

Anpassen der Meldung "Erfolgreich/Fehler" für Benutzer

Wenn Sie über Befehle verfügen:

```
auth-prompt accept "GOOD_AUTH"
auth-prompt reject "BAD_AUTH"
```

Benutzer sehen diese Sequenz bei fehlgeschlagener/erfolgreicher Anmeldung über PIX:

```
PIX_PIX_PIX
 Username: asjdkl
 Password:
 "BAD_AUTH"
 "PIX_PIX_PIX"
 Username: cse
 Password:
```

```
"GOOD_AUTH"
```

Leerlauf- und absolute Timeouts pro Benutzer

Leerlauf- und absolute uauth-Timeouts können pro Benutzer vom TACACS+-Server abgemeldet werden. Wenn alle Benutzer in Ihrem Netzwerk die gleiche "Timeout-Auth" haben sollen, implementieren Sie dies nicht! Wenn Sie jedoch unterschiedliche uauths pro Benutzer benötigen, lesen Sie weiter.

In diesem Beispiel wird der Befehl **timeout uauth 3:00:00** verwendet. Wenn sich eine Person authentifiziert hat, muss sie sich nicht drei Stunden lang erneut authentifizieren. Wenn Sie jedoch einen Benutzer mit diesem Profil einrichten und die TACACS-AAA-*Autorisierung* im PIX aktiviert haben, überschreiben die Leerlauf- und absoluten Timeouts im Benutzerprofil die Timeout-Autorisierung im PIX für diesen Benutzer. Dies bedeutet nicht, dass die Telnet-Sitzung über den PIX nach dem Leerlauf-/absoluten Timeout getrennt wird. Es kontrolliert lediglich, ob eine erneute Authentifizierung erfolgt.

Dieses Profil stammt von der Freeware TACACS+:

```
user = timeout {
default service = permit
login = cleartext "timeout"
service = exec {
timeout = 2
idletime = 1
}
}
```

Führen Sie nach der Authentifizierung einen Befehl **show uauth** auf dem PIX aus:

```
pix-5# show uauth
                        Current    Most Seen
 Authenticated Users       1          1
 Authen In Progress        0          1
 user 'timeout' at 10.31.1.5, authorized to:
    port 11.11.11.15/telnet
    absolute   timeout: 0:02:00
    inactivity timeout: 0:01:00
```

Nachdem der Benutzer eine Minute lang im Leerlauf sitzt, wird das Debuggen auf dem PIX angezeigt:

```
109012: Authen Session End: user 'timeout', Sid 19, elapsed 91 seconds
```

Der Benutzer muss sich erneut authentifizieren, wenn er zum gleichen Zielhost oder zu einem anderen Host zurückkehrt.

Virtuelles HTTP

Wenn an Standorten außerhalb des PIX sowie auf dem PIX selbst eine Authentifizierung erforderlich ist, kann gelegentlich ein ungewöhnliches Browserverhalten beobachtet werden, da Browser den Benutzernamen und das Kennwort zwischenspeichern.

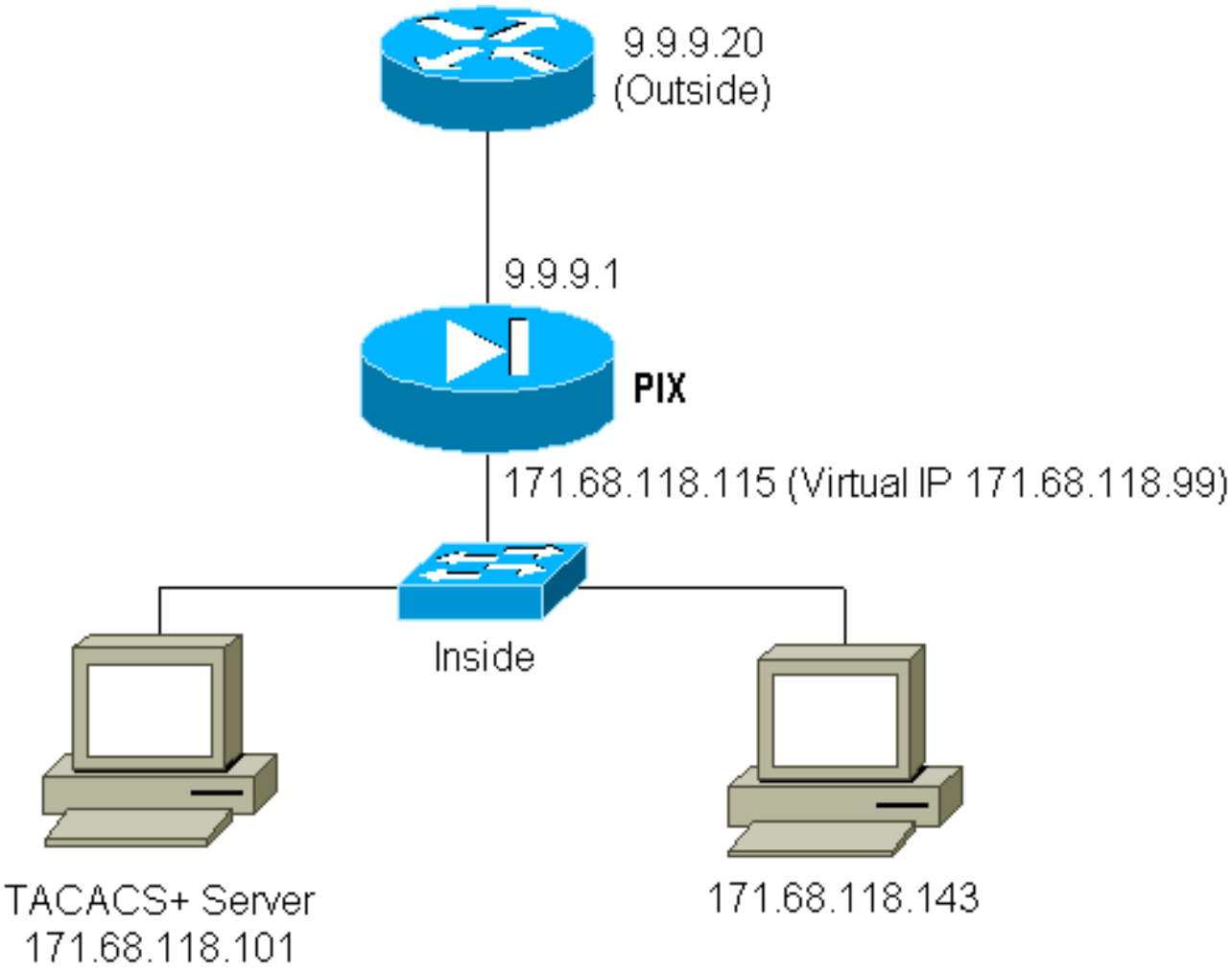
Um dies zu vermeiden, können Sie virtuelles HTTP implementieren, indem Sie der PIX-Konfiguration mithilfe des folgenden Befehls eine RFC 1918 -Adresse (eine Adresse, die im

Internet nicht routbar ist, aber für das PIX-interne Netzwerk gültig und eindeutig ist) hinzufügen:

```
virtual http #.#.#.# [warn]
```

Wenn der Benutzer versucht, den PIX zu verlassen, ist eine Authentifizierung erforderlich. Wenn der Warn-Parameter vorhanden ist, erhält der Benutzer eine Umleitungsmeldung. Die Authentifizierung ist für die Dauer der Authentifizierung gut. Legen Sie, wie in der Dokumentation angegeben, bei virtuellem HTTP nicht die Dauer des **Timeout-**Befehls auf 0 Sekunden fest. Dadurch werden HTTP-Verbindungen zum echten Webserver verhindert.

Ausgehendes Diagramm für virtuelles HTTP



PIX-Konfiguration Virtual HTTP Outbound

```
ip address outside 9.9.9.1 255.255.255.0
 ip address inside 171.68.118.115 255.255.255.0
 global (outside) 1 9.9.9.5-9.9.9.9 netmask 255.0.0.0
 timeout uauth 01:00:00
 aaa-server TACACS+ protocol tacacs+
 aaa-server AuthOutbound protocol tacacs+
 aaa-server AuthOutbound (inside) host 171.68.118.101 cisco timeout 10
 aaa authentication any outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthOutbound
 virtual http 171.68.118.99
```

```
auth-prompt THIS_IS_PIX_5
```

Virtuelles Telnet

Es ist zwar möglich, den PIX so zu konfigurieren, dass der gesamte ein- und ausgehende Datenverkehr authentifiziert wird, dies ist jedoch nicht empfehlenswert. Dies liegt daran, dass einige Protokolle wie "mail" nicht einfach authentifiziert werden können. Wenn ein Mailserver und ein Client versuchen, über das PIX zu kommunizieren, wenn der gesamte Datenverkehr über das PIX authentifiziert wird, zeigt das PIX-Syslog für nicht authentifizierbare Protokolle Meldungen wie:

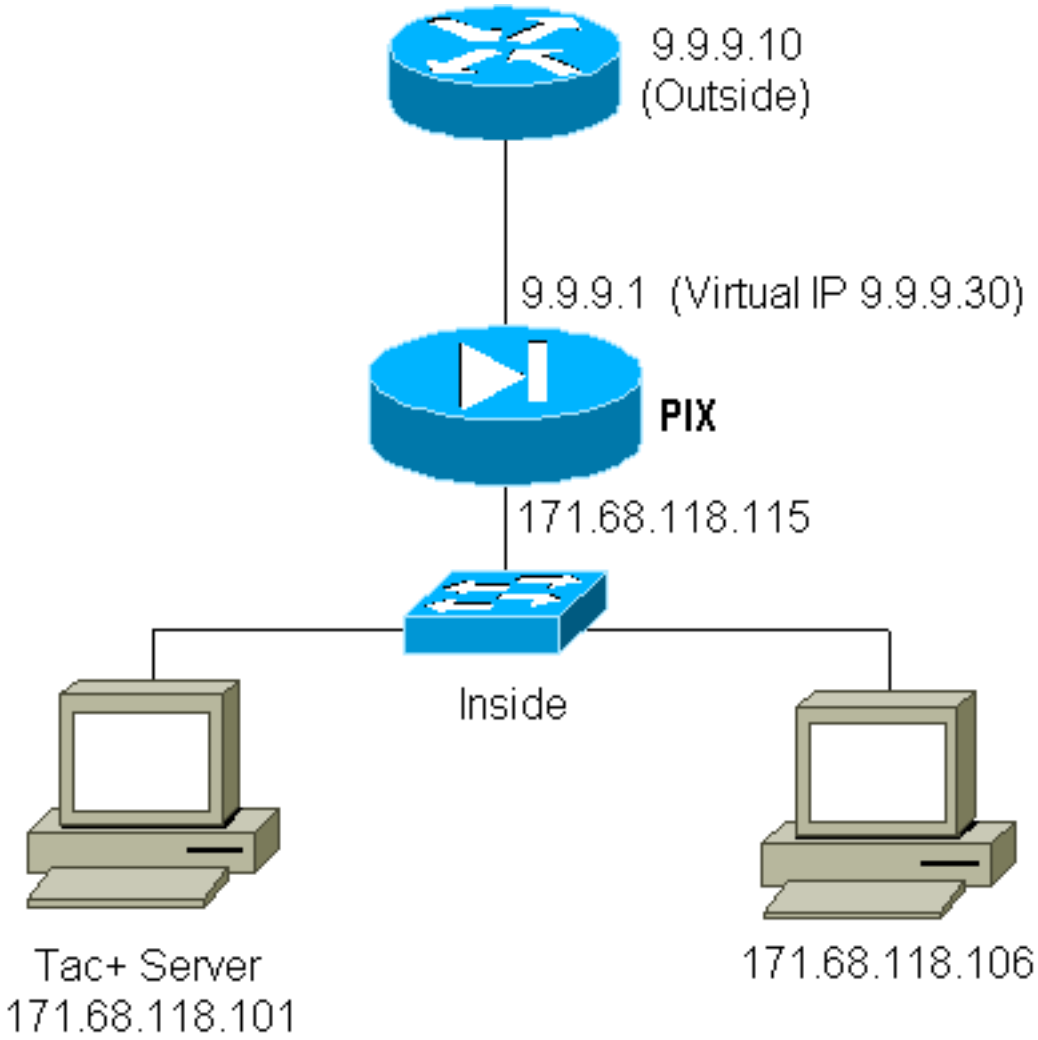
```
109001: Auth start for user '???' from 9.9.9.10/11094
    to 171.68.118.106/25
 109009: Authorization denied from 171.68.118.106/49 to
    9.9.9.10/11094 (not authenticated
```

Da E-Mail und einige andere Dienste nicht interaktiv genug sind, um sich zu authentifizieren, besteht eine Lösung darin, den Befehl **außer** dem Befehl für Authentifizierung/Autorisierung zu verwenden (alle authentifizieren, außer für Quelle/Ziel des Mailservers/Clients).

Wenn ein ungewöhnlicher Dienst wirklich authentifiziert werden muss, kann dies mithilfe des **virtuellen telnet-**Befehls geschehen. Dieser Befehl ermöglicht die Authentifizierung der virtuellen Telnet-IP. Nach dieser Authentifizierung kann der Datenverkehr für den ungewöhnlichen Dienst an den echten Server weitergeleitet werden.

In diesem Beispiel soll der TCP-Port 49-Datenverkehr vom externen Host 9.9.9.10 zum internen Host 171.68.118.106 fließen. Da dieser Datenverkehr nicht wirklich authentifizierbar ist, richten wir ein virtuelles Telnet ein. Für ein virtuelles eingehendes Telnet muss ein statisches Gerät zugeordnet sein. Hier sind sowohl 9.9.9.20 als auch 171.68.118.20 virtuelle Adressen.

Diagramm für eingehenden virtuellen Telnet-Datenverkehr



PIX-Konfiguration Virtual Telnet Inbound

```
ip address outside 9.9.9.1 255.255.255.0
 ip address inside 171.68.118.115 255.255.255.0
 static (inside,outside) 9.9.9.20 171.68.118.20 netmask 255.255.255.255 0 0
 static (inside,outside) 9.9.9.30 171.68.118.106 netmask 255.255.255.255 0 0
 conduit permit tcp host 9.9.9.20 eq telnet any
 conduit permit tcp host 9.9.9.30 eq tacacs any
 aaa-server TACACS+ protocol tacacs+
 aaa-server AuthInbound protocol tacacs+
 aaa-server AuthInbound (inside) host 171.68.118.101 cisco timeout 5
 AAA authentication any inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
 virtual telnet 9.9.9.20
```

TACACS+-Serverbenutzerkonfiguration Virtual Telnet Inbound

```
user = pinecone {
default service = permit
        login = cleartext "pinecone"
service = exec {
        timeout = 10
        idletime = 10
        }
}
```

PIX Debug Virtual Telnet Inbound

Der Benutzer mit der Adresse 9.9.9.10 muss sich zuerst durch Telnetting an die Adresse 9.9.9.20 auf dem PIX authentifizieren:

```
pixfirewall# 109001: Auth start for user '???' from 9.9.9.10/11099
    to 171.68.118.20/23
 109011: Authen Session Start: user 'pinecone', Sid 13
 109005: Authentication succeeded for user 'pinecone'
    from 171.68.118.20/23 to 9.9.9.10/1470
```

Nach erfolgreicher Authentifizierung zeigt der Befehl **show uauth**, dass der Benutzer die Zeit auf dem Messgerät hat:

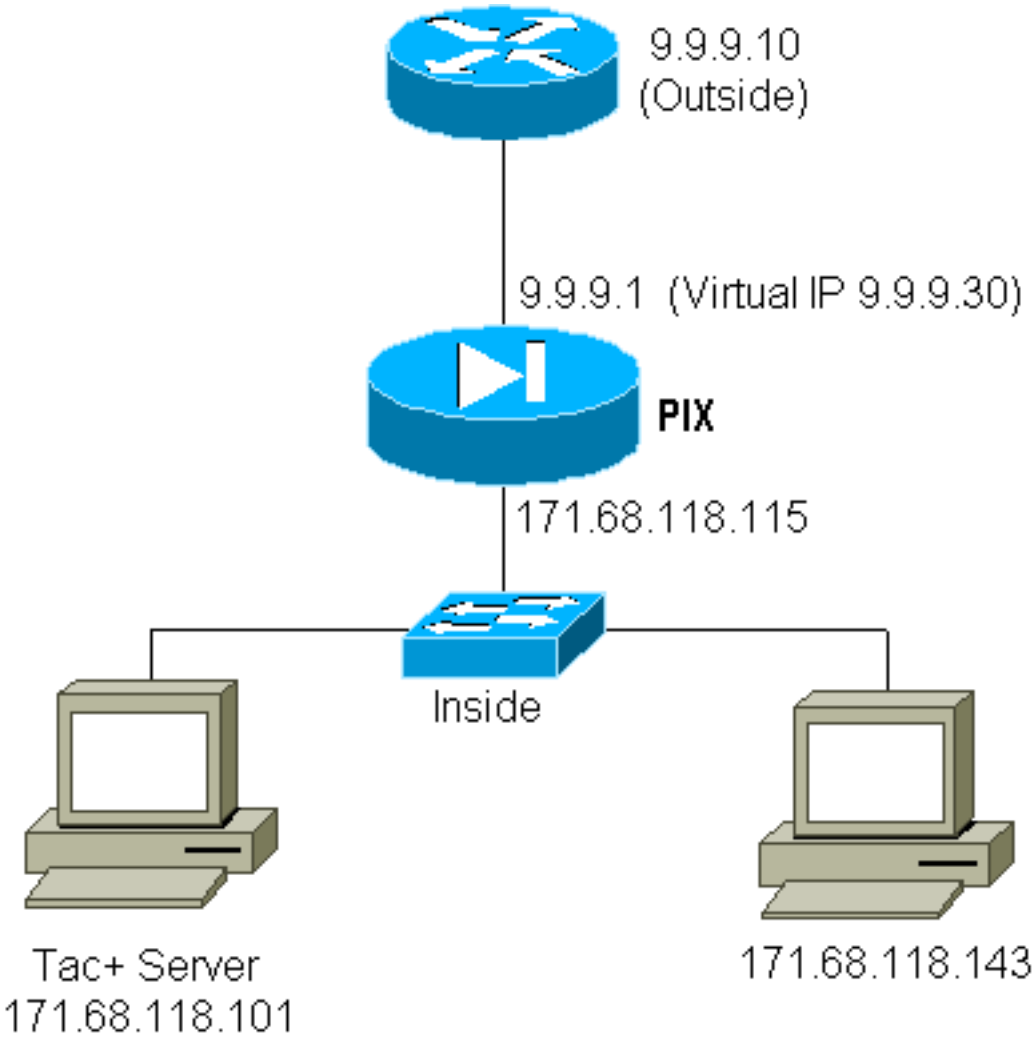
```
pixfirewall# show uauth
                        Current    Most Seen
 Authenticated Users       1          1
 Authen In Progress        0          1
 user 'pinecone' at 9.9.9.10, authenticated
    absolute   timeout: 0:10:00
    inactivity timeout: 0:10:00
```

Hier möchte das Gerät unter 9.9.9.10 TCP/49-Datenverkehr an das Gerät unter 171.68.118.106 senden:

```
pixfirewall# 109001: Auth start for user 'pinecone' from 9.9.9.10/11104
     to 171.68.118.20/23
 109011: Authen Session Start: user 'pinecone', Sid 14
 109005: Authentication succeeded for user 'pinecone' from 171.68.118.20/23
     to 9.9.9.10/1470
 302001: Built TCP connection 23 for faddr 9.9.9.10/11104 gaddr 9.9.9.30/49
  laddr 171.68.118.106/49 (pinecone)
 302002: Teardown TCP connection 23 faddr 9.9.9.10/11104 gaddr 9.9.9.30/49
  laddr 171.68.118.106/49 duration 0:00:10 bytes 179 (pinecone)
```

Virtuelles Telnet - Ausgehend

Da ausgehender Datenverkehr standardmäßig zulässig ist, ist für die Verwendung von ausgehenden virtuellen Telnet-Verbindungen kein statisches Gerät erforderlich. In diesem Beispiel authentifiziert der interne Benutzer unter 171.68.118.143 Telnet-to-virtual 9.9.9.30 und authentifiziert sich. Die Telnet-Verbindung wird sofort getrennt. Nach der Authentifizierung ist TCP-Datenverkehr vom 171.68.118.143 zum Server unter 9.9.9.10 zulässig:



PIX-Konfiguration Virtual Telnet Outbound

```
ip address outside 9.9.9.1 255.255.255.0
 ip address inside 171.68.118.115 255.255.255.0
 global (outside) 1 9.9.9.5-9.9.9.9 netmask 255.0.0.0
 timeout uauth 00:05:00
 aaa-server TACACS+ protocol tacacs+
 aaa-server AuthOutbound protocol tacacs+
 aaa-server AuthOutbound (inside) host 171.68.118.101 cisco timeout 10
 AAA authentication any outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthOutbound
 virtual telnet 9.9.9.30
```

PIX Debug Virtual Telnet Outbound

```
109001: Auth start for user '???' from 171.68.118.143/1536
     to 9.9.9.30/23
 109011: Authen Session Start: user 'timeout_143', Sid 25
 109005: Authentication succeeded for user 'timeout_143' from
    171.68.118.143/1536 to 9. 9.9.30/23
 302001: Built TCP connection 46 for faddr 9.9.9.10/80 gaddr
    9.9.9.30/1537 laddr 171.68 .118.143/1537 (timeout_143)
 304001: timeout_143@171.68.118.143 Accessed URL 9.9.9.10:/
 302001: Built TCP connection 47 for faddr 9.9.9.10/80 gaddr
    9.9.9.30/1538 laddr 171.68 .118.143/1538 (timeout_143)
 302002: Teardown TCP connection 46 faddr 9.9.9.10/80 gaddr
    9.9.9.30/1537 laddr 171.68. 118.143/1537 duration 0:00:03
```

```
   bytes 625 (timeout_143)
304001: timeout_143@171.68.118.143 Accessed URL 9.9.9.10:/
302002: Teardown TCP connection 47 faddr 9.9.9.10/80 gaddr
   9.9.9.30/1538 laddr 171.68. 118.143/1538 duration 0:00:01
   bytes 2281 (timeout_143)
302009: 0 in use, 1 most used
```

Logout für virtuelles Telnet

Wenn der Benutzer Telnet zur virtuellen Telnet-IP-Adresse wechselt, wird der Befehl **show uauth** angezeigt.

Wenn der Benutzer verhindern möchte, dass Datenverkehr nach Beendigung der Sitzung weitergeleitet wird (wenn noch Zeit in der Warteschlange verbleibt), muss der Benutzer erneut Telnet zur virtuellen Telnet-IP-Verbindung nutzen. Dadurch wird die Sitzung deaktiviert.

Port-Autorisierung

Sie können eine Autorisierung für eine Reihe von Ports benötigen. In diesem Beispiel war noch eine Authentifizierung für alle ausgehenden Datenverkehr erforderlich, jedoch war nur eine Autorisierung für die TCP-Ports 23-49 erforderlich.

PIX-Konfiguration

```
AAA authentication any outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthOutbound
AAA authorization tcp/23-49 outbound 0.0.0.0 0.0.0.0
   0.0.0.0 0.0.0.0 AuthOutbound
```

Wenn das Telnet zwischen 171.68.118.143 und 9.9.9.10 betrieben wurde, fanden Authentifizierung und Autorisierung statt, da der Telnet-Port 23 im Bereich von 23 bis 49 liegt.

Wenn eine HTTP-Sitzung zwischen 171.68.118.143 und 9.9.9.10 stattfindet, müssen Sie sich noch authentifizieren, aber der PIX bittet den TACACS+-Server nicht, HTTP zu autorisieren, da 80 nicht im 23-49-Bereich liegt.

TACACS+ Freeware Server-Konfiguration

```
user = telnetrange {
        login = cleartext "telnetrange"
        cmd = tcp/23-49 {
        permit 9.9.9.10
        }
        }
```

Beachten Sie, dass der PIX "`cmd=tcp/23-49`" und "`cmd-arg=9.9.9.10`" an den TACACS+-Server sendet.

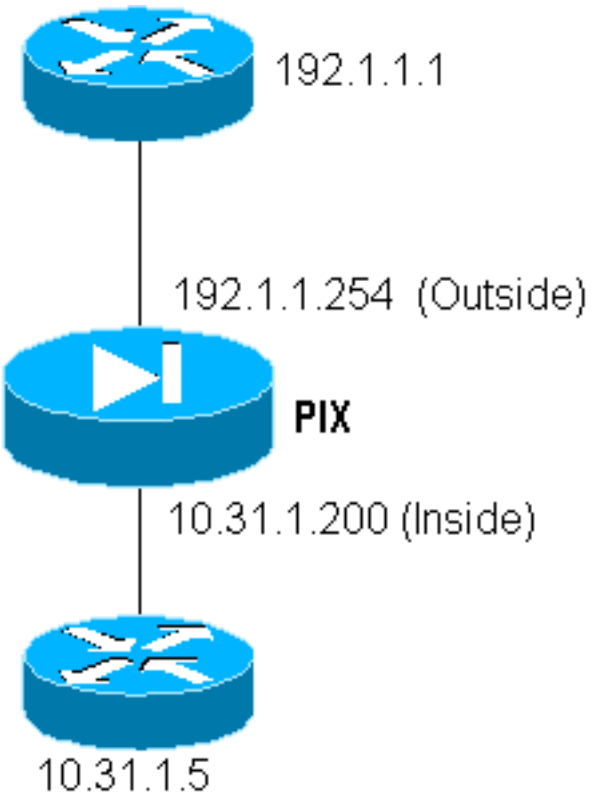
Debuggen auf dem PIX

```
109001: Auth start for user '???' from 171.68.118.143/1051
```

```
   to 9.9.9.10/23
109011: Authen Session Start: user 'telnetrange', Sid 0
109005: Authentication succeeded for user 'telnetrange'
   from 171.68.118.143/1051 to 9. 9.9.10/23
109011: Authen Session Start: user 'telnetrange', Sid 0
109007: Authorization permitted for user 'telnetrange'
   from 171.68.118.143/1051 to 9.9 .9.10/23
302001: Built TCP connection 0 for faddr 9.9.9.10/23
   gaddr 9.9.9.5/1051 laddr 171.68.1 18.143/1051 (telnetrange)
109001: Auth start for user '???' from 171.68.118.143/1105
   to 9.9.9.10/80
109001: Auth start for user '???' from 171.68.118.143/1110
   to 9.9.9.10/80
109011: Authen Session Start: user 'telnetrange', Sid 1
109005: Authentication succeeded for user 'telnetrange'
   from 171.68.118.143/1110 to 9. 9.9.10/80
302001: Built TCP connection 1 for faddr 9.9.9.10/80 gaddr 9.9.9.5/1110
 laddr 171.68.1 18.143/1110 (telnetrange)
302001: Built TCP connection 2 for faddr 9.9.9.10/80 gaddr 9.9.9.5/1111
 laddr 171.68.1 18.143/1111 (telnetrange)
302002: Teardown TCP connection 1 faddr 9.9.9.10/80 gaddr 9.9.9.5/1110
 laddr 171.68.11 8.143/1110 duration 0:00:08 bytes 338 (telnetrange)
304001: timeout_143@171.68.118.143 Accessed URL 9.9.9.10:/
302002: Teardown TCP connection 2 faddr 9.9.9.10/80 gaddr 9.9.9.5/1111
 laddr 171.68.11 8.143/1111 duration 0:00:01 bytes 2329 (telnetrange)
```

AAA-Abrechnung für Datenverkehr außer HTTP, FTP und Telnet

PIX Software Version 5.0 ändert die Traffic Accounting-Funktion. Die Accounting-Datensätze können nach Abschluss der Authentifizierung für anderen Datenverkehr als HTTP, FTP und Telnet abgeschnitten werden.



Um eine Datei vom externen Router (192.1.1.1) auf den internen Router (10.31.1.5) zu kopieren, fügen Sie virtuelle Telnet hinzu, um eine Lücke für den TFTP-Prozess zu öffnen:

```
virtual telnet 192.1.1.30
static (inside,outside) 192.1.1.30 10.31.1.5 netmask 255.255.255.255 0 0
conduit permit udp any any
AAA authentication udp/69 inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
AAA authorization udp/69 inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
AAA accounting udp/0 inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 AuthInbound
```

Anschließend wird Telnet vom externen Router unter 192.1.1.1 zum virtuellen IP 192.1.1.30 übertragen und an die virtuelle Adresse authentifiziert, über die das UDP den PIX durchlaufen kann. In diesem Beispiel wurde der **Kopieen-TFTP-**Prozess von außen nach innen gestartet:

```
302006: Teardown UDP connection for faddr 192.1.1.1/7680
    gaddr 192.1.1.30/69 laddr 10.31.1.5/69
```

Für jede **Kopie des TFTP-Flash-Speichers** auf dem PIX (während dieser IOS-Kopie gab es drei) wird ein Accounting-Datensatz abgeschnitten und an den Authentifizierungsserver gesendet. Das nachfolgende Beispiel zeigt einen TACACS-Datensatz unter Cisco Secure Windows):

```
Date,Time,Username,Group-Name,Caller-Id,Acct-Flags,elapsed_time,
    service,bytes_in,bytes_out,paks_in,paks_out,
    task_id,addr,NAS-Portname,NAS-IP-Address,cmd
 04/28/2000,03:08:26,pixuser,Default Group,192.1.1.1,start,,,,,,,
    0x3c,,PIX,10.31.1.200,udp/69
```

Zugehörige Informationen

- PIX-Befehlsreferenz
- PIX-Produktsupport-Seite