Beispielkonfigurationen für PIX, TACACS+ und RADIUS: 4,4 x

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 an PIX 4.4.1 Authentication, Authorization, Accounting (AAA) gegenüber der vorherigen Version umfassen: AAA-Servergruppen und -Failover, Authentifizierung für den Zugriff auf die Aktivierungs- und serielle Konsole sowie die Annahme und Ablehnung von Aufforderungsmeldungen.

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 ungültig.

Angenommen, Sie haben 100 Benutzer im Netzwerk und möchten nur 6 dieser Benutzer in der Lage sein, FTP, Telnet oder HTTP außerhalb des Netzwerks zu betreiben. Sie weisen den PIX an, ausgehenden Datenverkehr zu authentifizieren und alle 6 Benutzer-IDs auf dem TACACS+/RADIUS-Sicherheitsserver anzugeben. Mit einfacher Authentifizierung können diese 6 Benutzer mit Benutzername und Kennwort authentifiziert werden, bevor sie die Authentifizierung beenden. Die anderen 94 Benutzer konnten nicht ausgehen. Das PIX fordert Benutzer zur Eingabe von Benutzername/Kennwort auf, gibt dann ihren Benutzernamen und ihr Kennwort an den TACACS+/RADIUS-Sicherheitsserver weiter. Je nach Antwort wird die Verbindung geöffnet oder verweigert. Diese 6 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 die Autorisierung hinzugefügt werden muss, d. h., dass die Benutzer autorisiert werden können, und nicht nur authentifiziert werden müssen, wer sie sind. Wenn wir eine Autorisierung zum PIX hinzufügen, sendet das PIX zunächst Terrys Benutzernamen und Kennwort an den Sicherheitsserver und

sendet dann eine Autorisierungsanfrage, in der dem Sicherheitsserver mitgeteilt wird, welchen "Befehl" Terry zu tun versucht. Wenn der Server korrekt eingerichtet ist, kann Terry "FTP 1.2.3.4" verwenden, aber es würde ihm die Möglichkeit verwehrt, überall HTTP oder Telnet zu nutzen.

Was der Benutzer mit Authentifizierung/Autorisierung auf

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

- **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 - Im Browser wird ein Fenster angezeigt, in dem Benutzername und Kennwort angefordert werden. 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 scheinbar "normal" funktionieren, HTTP-Verbindungen jedoch nicht, ist dies der Grund dafür.

Für alle Szenarien verwendete Sicherheitsserver-Konfigurationen

CiscoSecure 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 .*
}
}
}
```

Konfiguration des CiscoSecure UNIX RADIUS-Servers

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

```
user=adminuser {
radius=Cisco {
check_items= {
2="all"
}
reply_attributes= {
6=6
}
}
```

CiscoSecure NT 2.x RADIUS

Führen Sie diese Schritte aus.

1. Rufen Sie ein Kennwort im Abschnitt GUI der 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 eine Autorisierung hinzuzufügen, klicken Sie unten im Gruppensetup auf **Nicht übereinstimmende IOS-Befehle verweigern**.
3. Wählen Sie für jeden Befehl, den Sie zulassen möchten (z. B. Telnet) den 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 und/oder FTP).
7. Fügen Sie die PIX-IP im Abschnitt "GUI der NAS-Konfiguration" hinzu.

CiscoSecure 2.x TACACS+

Der Benutzer erhält ein Kennwort im Abschnitt User Setup (Benutzereinrichtung) der GUI.

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 **Add/Edit** für jeden Befehl aus, den Sie zulassen möchten (z. B. Telnet).
4. Wenn Sie Telnet für bestimmte Standorte zulassen möchten, geben Sie die IP(s) zur Genehmigung im Argumentrechteck ein (z. B. "permit 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 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.

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
```

TACACS+ Freeware Server-Konfiguration

```
key = "cisco"

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

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

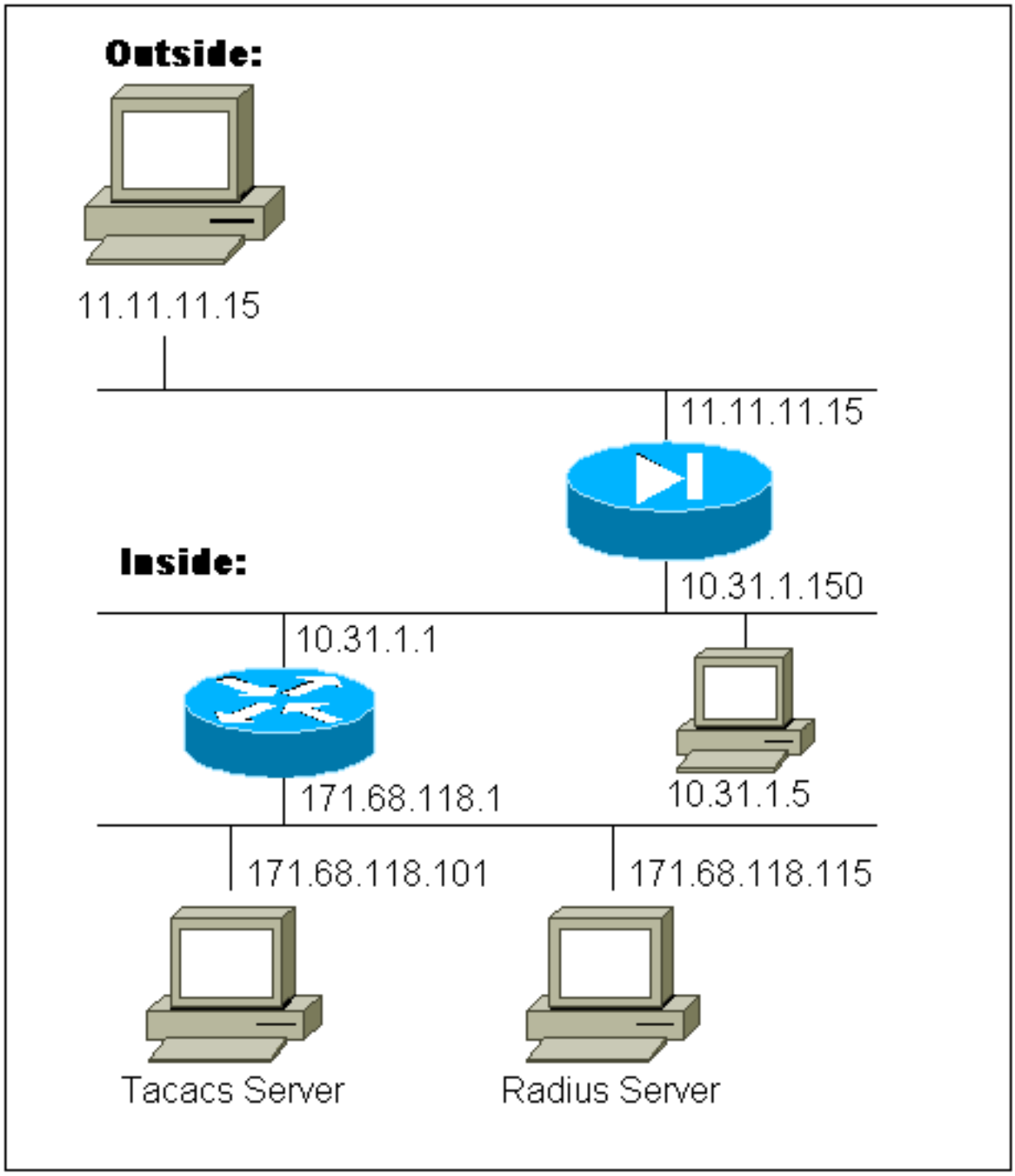
```
user = httponly {
login = cleartext "httponly"
cmd = http {
permit .*
}
}

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

Debugschritte

- Stellen Sie sicher, dass die PIX-Konfigurationen funktionieren, bevor Sie AAA (Authentication, Authorization, Accounting) 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 Befehl **zum Debuggen der Protokollierungskonsole** sollte auf einem stark ausgelasteten System nicht 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
 Building configuration...
 : Saved
 :
 PIX Version 4.4(1)
 nameif ethernet0 outside security0
 nameif ethernet1 inside security100
 nameif ethernet2 pix/intf2 security10
 nameif ethernet3 pix/intf3 security15
 enable password 8Ry2YjIyt7RRXU24 encrypted
 passwd 2KFQnbNIdI.2KYOU encrypted
 hostname pix-5
 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
 pager lines 24
 no logging timestamp
 logging console debugging
 no logging monitor
 no logging buffered
 logging trap debugging
 logging facility 20
 interface ethernet0 auto
 interface ethernet1 auto
 interface ethernet2 auto
 interface ethernet3 auto
 mtu outside 1500
 mtu inside 1500
 mtu pix/intf2 1500
 mtu pix/intf3 1500
 ip address outside 11.11.11.1 255.255.255.0
 ip address inside 10.31.1.150 255.255.255.0
 ip address pix/intf2 127.0.0.1 255.255.255.255
 ip address pix/intf3 127.0.0.1 255.255.255.255
 no failover
 failover timeout 0:00:00
 failover ip address outside 0.0.0.0
 failover ip address inside 0.0.0.0
 failover ip address pix/intf2 0.0.0.0
 failover ip address pix/intf3 0.0.0.0
 arp timeout 14400
 global (outside) 1 11.11.11.10-11.11.11.14 netmask
255.255.255.0
 static (inside,outside) 11.11.11.20 171.68.118.115
netmask 255.255.255.255 0 0
 static (inside,outside) 11.11.11.21 171.68.118.101
netmask 255.255.255.255 0 0
 static (inside,outside) 11.11.11.22 10.31.1.5 netmask
255.255.255.255 0 0
 conduit permit icmp any any
 conduit permit tcp any any
 no rip outside passive
 no rip outside default
 no rip inside passive
 no rip inside default
 no rip pix/intf2 passive
 no rip pix/intf2 default
 no rip pix/intf3 passive
 no rip pix/intf3 default
 route inside 0.0.0.0 0.0.0.0 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
!
!--- For any given list, multiple AAA servers can !---
be configured. They will be !--- tried sequentially if
any one of them is down. ! aaa-server Outgoing protocol
tacacs+ aaa-server Outgoing (inside) host 171.68.118.101
cisco timeout 10 aaa-server Incoming protocol radius
aaa-server Incoming (inside) host 171.68.118.115 cisco
timeout 10 aaa authentication ftp outbound 0.0.0.0
0.0.0.0 0.0.0.0 0.0.0.0 Outgoing aaa authentication http
outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 Outgoing aaa
```

```
authentication telnet outbound 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 Outgoing aaa authentication ftp inbound 0.0.0.0
0.0.0.0 0.0.0.0 0.0.0.0 Incoming aaa authentication http
inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 Incoming aaa
authentication telnet inbound 0.0.0.0 0.0.0.0 0.0.0.0
0.0.0.0 Incoming no snmp-server location no snmp-server
contact snmp-server community public no snmp-server
enable traps telnet timeout 5 terminal width 80
Cryptochecksum:b287a37a676262275a4201cac52399ca : end
```

Authentifizierungs-Debug-Beispiele aus PIX

In diesen Debugbeispielen:

Ausgehend

Der interne Benutzer mit der Adresse 10.31.1.5 initiiert Datenverkehr nach außerhalb von 11.11.11.15 und wird über TACACS+ authentifiziert (ausgehender Datenverkehr verwendet Serverliste "Outgoing", die den TACACS-Server 171.68.118.101 enthält).

Eingehend

Externer Benutzer mit der Nummer 11.11.11.15 initiiert Datenverkehr nach innen 10.31.1.5 (11.11.11.22) und wird über RADIUS authentifiziert (eingehender Datenverkehr verwendet die Serverliste "Incoming", die den RADIUS-Server 171.68.115 umfasst).

PIX Debug - Gute Authentifizierung - TACACS+

Im folgenden Beispiel wird das PIX-Debuggen mit guter Authentifizierung veranschaulicht:

```
109001: Auth start for user '???' from 10.31.1.5/11004 to 11.11.11.15/23
 109011: Authen Session Start: user 'ddunlap', sid 3
 109005: Authentication succeeded for user 'ddunlap'
 from 10.31.1.5/11004 to 11.11.11.15/23
 109012: Authen Session End: user 'ddunlap', sid 3, elapsed 1 seconds
 302001: Built outbound TCP connection 4 for faddr 11.11.11.15/23 gaddr
11.11.11.22/11004 laddr 10.31.1.5/11004
```

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

Im folgenden Beispiel wird das PIX-Debuggen mit fehlerhafter Authentifizierung (Benutzername oder Kennwort) veranschaulicht. Der Benutzer sieht vier Benutzername/Kennwort-Sets. Die folgende Meldung wird angezeigt: "Fehler: maximale Anzahl von Versuchen überschritten."

```
109001: Auth start for user '???' from 10.31.1.5/11005 to 11.11.11.15/23
 109006: Authentication failed for user '' from 10.31.1.5/11005 to 11.11.11.15/23
```

PIX-Debuggen - Senden Sie Ping, aber keine Antwort - TACACS+

Das nachfolgende Beispiel zeigt das PIX-Debuggen für einen Pingable-Server, der nicht mit dem PIX kommuniziert. Der Benutzer sieht den Benutzernamen einmal, und PIX fragt nie nach einem Kennwort (dies ist auf Telnet).

```
'Error: Max number of tries exceeded'
 109001: Auth start for user '???' from 10.31.1.5/11006 to 11.11.11.15/23
 109002: Auth from 10.31.1.5/11006 to 11.11.11.15/23 failed
 (server 171.68.118.101 failed)
 109002: Auth from 10.31.1.5/11006 to 11.11.11.15/23 failed
 (server 171.68.118.101 failed)
 304006: URL Server 171.68.118.101 not responding, trying 171.68.118.101
 109002: Auth from 10.31.1.5/11006 to 11.11.11.15/23 failed
 (server 171.68.118.101 failed)
 109006: Authentication failed for user '' from 10.31.1.5/11006 to 11.11.11.15/23
```

PIX Debug - Can't Ping Server - TACACS+

Im folgenden Beispiel wird das PIX-Debuggen für einen Server veranschaulicht, der nicht pingfähig ist. Der Benutzer sieht den Benutzernamen einmal. PIX fragt nie nach einem Passwort (dies ist auf Telnet). Die folgende Meldung wird angezeigt: "Timeout to TACACS+ server" und "Fehler: Maximale Anzahl der überschritten" (die Konfiguration in diesem Beispiel spiegelt einen Scheinserver wider).

```
109001: Auth start for user '???' from 10.31.1.5/11007 to 11.11.11.15/23
 109002: Auth from 10.31.1.5/11007 to 11.11.11.15/23 failed
 (server 171.68.118.199 failed)
 109002: Auth from 10.31.1.5/11007 to 11.11.11.15/23 failed
 (server 171.68.118.199 failed)
 304006: URL Server 171.68.118.199 not responding, trying 171.68.118.199
 109002: Auth from 10.31.1.5/11007 to 11.11.11.15/23 failed
 (server 171.68.118.199 failed)
 109006: Authentication failed for user '' from 10.31.1.5/11007 to 11.11.11.15/23
```

PIX Debug - Gute Authentifizierung - RADIUS

Im folgenden Beispiel wird das PIX-Debuggen mit guter Authentifizierung veranschaulicht:

```
109001: Auth start for user '???' from 11.11.11.15/11003 to 10.31.1.5/23
 109011: Authen Session Start: user 'adminuser', sid 4
 109005: Authentication succeeded for user 'adminuser'
 from 10.31.1.5/23 to 11.11.11.15/11003
 109012: Authen Session End: user 'adminuser', sid 4, elapsed 1 seconds
 302001: Built inbound TCP connection 5 for faddr
 11.11.11.15/11003 gaddr 11.11.11.22/23 laddr 10.31.1.5/23
```

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

Im folgenden Beispiel wird das PIX-Debuggen mit fehlerhafter Authentifizierung (Benutzername oder Kennwort) veranschaulicht. Der Benutzer erhält eine Anfrage für Benutzername und Kennwort. Wenn eine der Antworten falsch ist, wird die Meldung "Falsches Kennwort" viermal angezeigt. Anschließend wird die Verbindung zum Benutzer getrennt. Diesem Problem wurde die Bug-ID #CSCdm46934 zugewiesen.

```
'Error: Max number of tries exceeded'
 109001: Auth start for user '???' from 11.11.11.15/11007 to 10.31.1.5/23
 109006: Authentication failed for user '' from 10.31.1.5/23 to 11.11.11.15/11007
```

PIX Debug - Deamon Down, Kommunikation mit PIX - RADIUS nicht möglich

Das nachfolgende Beispiel zeigt das PIX-Debuggen mit einem Pingable-Server, der Daemon ist jedoch ausgefallen. Der Server kommuniziert nicht mit PIX. Der Benutzer sieht Benutzername und anschließend Kennwort. Folgende Meldungen werden angezeigt: "RADIUS-Server fehlgeschlagen" und "Fehler: Max. Anzahl von Versuchen überschritten".

```
109001: Auth start for user '???' from 11.11.11.15/11008 to 10.31.1.5/23
 109002: Auth from 10.31.1.5/23 to 11.11.11.15/11008 failed
 (server 171.68.118.115 failed)
 109002: Auth from 10.31.1.5/23 to 11.11.11.15/11008 failed
 (server 171.68.118.115 failed)
 304006: URL Server 171.68.118.115 not responding, trying 171.68.118.115
 109002: Auth from 10.31.1.5/23 to 11.11.11.15/11008 failed
 (server 171.68.118.115 failed)
 109006: Authentication failed for user '' from 10.31.1.5/23 to 11.11.11.15/11008
```

PIX Debug - Ping-Server oder Key/Client-Mismatch nicht möglich - RADIUS

Das nachfolgende Beispiel zeigt das PIX-Debuggen für einen Server, der nicht pingfähig ist oder bei dem eine Schlüssel-Client-Diskrepanz vorliegt. Der Benutzer sieht Benutzername und Kennwort. Folgende Meldungen werden angezeigt: "Timeout to RADIUS server" und "Fehler: Die maximale Anzahl der Versuche wurde überschritten" (der Server in der Konfiguration dient nur zum Beispiel).

```
109001: Auth start for user '???' from 11.11.11.15/11009 to 10.31.1.5/23
 109002: Auth from 10.31.1.5/23 to 11.11.11.15/11009 failed
 (server 171.68.118.199 failed)
 109002: Auth from 10.31.1.5/23 to 11.11.11.15/11009 failed
 (server 171.68.118.199 failed)
 304006: URL Server 171.68.118.199 not responding, trying 171.68.118.199
 109002: Auth from 10.31.1.5/23 to 11.11.11.15/11009 failed
 (server 171.68.118.199 failed)
 109006: Authentication failed for user '' from 10.31.1.5/23 to 11.11.11.15/11009
```

Autorisierung hinzufügen

Da die Autorisierung ohne Authentifizierung nicht gültig ist, ist eine Autorisierung für denselben Quell- und Zielbereich erforderlich:

```
aaa authorization ftp outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
aaa authorization http outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
aaa authorization telnet outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0
```

Ausgehend

Beachten Sie, dass keine Autorisierung für "incoming" hinzugefügt wird, da eingehender Datenverkehr mit RADIUS authentifiziert wird und die RADIUS-Autorisierung ungültig ist.

Debug-Beispiele für Authentifizierung und Autorisierung aus PIX

PIX-Debugging mit guter Authentifizierung und erfolgreicher Autorisierung - TACACS+

Das nachfolgende Beispiel zeigt das PIX-Debugging mit guter Authentifizierung und erfolgreicher Autorisierung:

```
109001: Auth start for user '???' from 10.31.1.5/11002 to 11.11.11.15/23
 109011: Authen Session Start: user 'can_only_do_telnet', sid 7
 109005: Authentication succeeded for user 'can_only_do_telnet'
 from 10.31.1.5/11002 to 11.11.11.15/23
 109011: Authen Session Start: user 'can_only_do_telnet', sid 7
 109007: Authorization permitted for user 'can_only_do_telnet'
 from 10.31.1.5/11002 to 11.11.11.15/23
 109012: Authen Session End: user 'can_only_do_telnet', sid 7,
 elapsed 1 seconds
 302001: Built outbound TCP connection 6 for faddr 11.11.11.15/23
 gaddr 11.11.11.22/11002 laddr 10.31.1.5/11002 (can_only_do_telnet)
```

PIX Debug - Gute Authentifizierung, fehlgeschlagene Autorisierung - TACACS+

Das nachfolgende Beispiel zeigt das PIX-Debuggen mit guter Authentifizierung, aber fehlgeschlagene Autorisierung:

Hier sieht der Benutzer auch die Meldung "Fehler: Autorisierung verweigert"

```
109001: Auth start for user '???' from 10.31.1.5/11000 to 11.11.11.15/23
 109011: Authen Session Start: user 'can_only_do_ftp', sid 5
 109005: Authentication succeeded for user 'can_only_do_ftp'
 from 10.31.1.5/11000 to 11.11.11.15/23
 109008: Authorization denied for user 'can_only_do_ftp' from
 10.31.1.5/11000 to 11.11.11.15/23
 109012: Authen Session End: user 'can_only_do_ftp', sid 5, elapsed 33 seconds
```

Hinzufügen von Buchhaltung

TACACS+

```
aaa accounting any outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 Outgoing
```

Debug sieht gleich aus, ob die Buchhaltung aktiviert oder deaktiviert ist. Zum Zeitpunkt des Built-Projekts wird jedoch ein "Start"-Accounting-Datensatz gesendet. Zum Zeitpunkt des "Teardown"-Verfahrens wird ein "Stopp"-Buchhaltungsdatensatz gesendet.

Die TACACS+-Accounting-Datensätze sehen wie folgt aus (diese stammen von CiscoSecure UNIX. die in CiscoSecure NT können stattdessen durch Kommas getrennt werden):

```
Thu Jun  3 10:41:50 1999 10.31.1.150 can_only_do_telnet
 PIX 10.31.1.5 start task_id=0x7 foreign_ip=11.11.11.15
local_ip=10.31.1.5  cmd=telnet
Thu Jun  3 10:41:55 1999 10.31.1.150 can_only_do_telnet PIX 10.31.1.5
 stop task_id=0x7 foreign_ip=11.11.11.15
local_ip=10.31.1.5 cmd=telnet elapsed_time=4 bytes_in=74 bytes_out=27
```

RADIUS

```
aaa accounting any inbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 Incoming
```

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

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

```
10.31.1.150adminuser -- start server=rtp-evergreen.rtp.cisco.com
time=14:53:11 date=06/3/1999 task_id=0x00000008
 Thu Jun  3 15:53:11 1999
         Acct-Status-Type = Start
         Client-Id = 10.31.1.150
         Login-Host = 10.31.1.5
         Login-TCP-Port = 23
         Acct-Session-Id = "0x00000008"
         User-Name = "adminuser"
 10.31.1.150 adminuser -- stop server=rtp-evergreen.rtp.cisco.com
time=14:54:24 date=06/ 3/1999 task_id=0x00000008
 Thu Jun  3 15:54:24 1999
         Acct-Status-Type = Stop
         Client-Id = 10.31.1.150
         Login-Host = 10.31.1.5
         Login-TCP-Port = 23
         Acct-Session-Id = "0x00000008"
         User-Name = "adminuser"
         Acct-Session-Time = 73
         Acct-Input-Octets = 27
         Acct-Output-Octets = 73
```

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 Folgendes tun:

```
aaa authentication except outbound 10.31.1.60 255.255.255.255
11.11.11.15 255.255.255.255 Outgoing
aaa authorization except outbound 10.31.1.60 255.255.255.255
11.11.11.15 255.255.255.255 Outgoing
```

Wenn Sie IP-Adressen aus der Authentifizierung "ausnehmen" und über eine Autorisierung verfügen, müssen Sie diese auch 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 (d. h. 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. Im folgenden Beispiel wird eine andere Netzwerkkonfiguration verwendet, die Konzepte sind jedoch identisch.

Der Benutzer meldet sich per Telnet über den PIX an und 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 seinem 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 jedoch aufgrund der Art des Protokolls anders. Unten sehen Sie 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. Zwischen dem Start- und dem Stopp-Datensatz lag weniger als eine Sekunde.) 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. Aufgrund der Art der HTTP-Verbindungen funktionieren jedoch keine Max-Sessions und keine Ansicht der angemeldeten Benutzer.

Authentifizierung und Aktivierung auf dem PIX selbst

Die vorige Diskussion betraf die Authentifizierung des Telnet- (und HTTP-, FTP-) Datenverkehrs über das PIX. Im folgenden Beispiel stellen wir sicher, dass Telnet zum pix ohne Authentifizierung funktioniert in:

```
telnet 10.31.1.5 255.255.255.255
passwd ww
```

Anschließend fügen wir den Befehl hinzu, um Benutzer zu authentifizieren Telnetting zu PIX:

```
aaa authentication telnet console Outgoing
```

Wenn Benutzer Telnet an den PIX anschließen, werden sie zur Eingabe des Telnet-Kennworts aufgefordert ("ww"). Der PIX fordert in diesem Fall auch TACACS+ (da die Serverliste "Ausgehend" verwendet wird) oder RADIUS-Benutzername und -Kennwort an.

```
aaa authentication enable console Outgoing
```

Mit diesem Befehl wird der Benutzer aufgefordert, einen Benutzernamen und ein Kennwort einzugeben, die an den TACACS- oder RADIUS-Server gesendet werden. Da in diesem Fall die Serverliste "Ausgehend" verwendet wird, wird die Anforderung an den TACACS-Server gesendet. Da das Authentifizierungspaket für "enable" mit dem Authentifizierungspaket für die Anmeldung identisch ist, kann der Benutzer über TACACS oder RADIUS mit demselben Benutzernamen/Kennwort aktivieren, vorausgesetzt, der Benutzer kann sich bei PIX mit TACACS oder RADIUS anmelden. Diesem Problem wurde die Bug-ID #CSCdm47044 zugewiesen.

Wenn der Server ausgefallen ist, kann der Benutzer auf den PIX-Aktivierungsmodus zugreifen, indem er "PIX" für den Benutzernamen und das normale enable-Kennwort vom PIX eingibt

("enable password any"). Wenn "enable password any" nicht in der PIX-Konfiguration enthalten ist, sollte der Benutzer "PIX" als Benutzernamen eingeben und die Eingabetaste drücken. Wenn das enable-Kennwort festgelegt, aber nicht bekannt ist, wird eine Kennwortwiederherstellungsdiskette benötigt, um zurückgesetzt zu werden.

Authentifizierung auf der seriellen Konsole

Der Befehl **aaa authentication serial console** erfordert eine Authentifizierungsüberprüfung, um auf die serielle Konsole des PIX zugreifen zu können. Wenn der Benutzer Konfigurationsbefehle über die Konsole ausführt, werden Syslog-Meldungen deaktiviert (wenn PIX so konfiguriert ist, dass Syslog auf der Debugebene an einen Syslog-Host gesendet wird). Im Folgenden sehen Sie ein Beispiel vom Syslog-Server:

```
Jun  5 07:24:09 [10.31.1.150.2.2] %PIX-5-111008: User 'cse' executed
the 'hostname' command.
```

Benutzer auffordern anzeigen

Wenn der Befehl vorhanden ist:

```
auth-prompt THIS_IS_PIX_5
```

Die Benutzer, die den PIX durchlaufen, sehen die Sequenz:

```
THIS_IS_PIX_5 [at which point one would enter the username]
 Password:[at which point one would enter the password]
```

und dann bei der Ankunft im ultimativen Zielfeld wird die Eingabeaufforderung "Benutzername:" und "Passwort:" angezeigt.

Diese Eingabeaufforderung wirkt sich nur auf Benutzer aus, die den PIX durchlaufen, nicht auf den PIX.

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

Anpassen der Meldung "Erfolgreich/Fehler" für Benutzer

Wenn die Befehle vorhanden sind:

```
auth-prompt accept "You're allowed through the pix"
auth-prompt reject "You blew it"
```

Bei fehlgeschlagener/erfolgreicher Anmeldung über PIX wird den Benutzern Folgendes angezeigt:

```
THIS_IS_PIX_5
 Username: asjdkl
```

```
Password:
"You blew it"
"THIS_IS_PIX_5"
Username: cse
Password:
"You're allowed through the pix"
```

Timeouts pro Benutzer bei Inaktivität und absoluten Zeitüberschreitungen

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, dann implementieren Sie dies nicht! Wenn Sie jedoch unterschiedliche uauths pro Benutzer benötigen, lesen Sie weiter.

In unserem Beispiel auf dem PIX verwenden wir den Befehl **timeout uauth 3:00:00**. Das bedeutet, dass sich eine Person nach ihrer Authentifizierung 3 Stunden lang nicht erneut authentifizieren muss. Wenn wir jedoch einen Benutzer mit dem folgenden Profil einrichten und die TACACS-AAA-Autorisierung im PIX aktiviert ist, überschreiben die Leerlauf- und absolute Zeitüberschreitungen 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. Sie kontrolliert lediglich, ob eine erneute Authentifizierung erfolgt oder nicht.

```
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 Ziel-Host 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

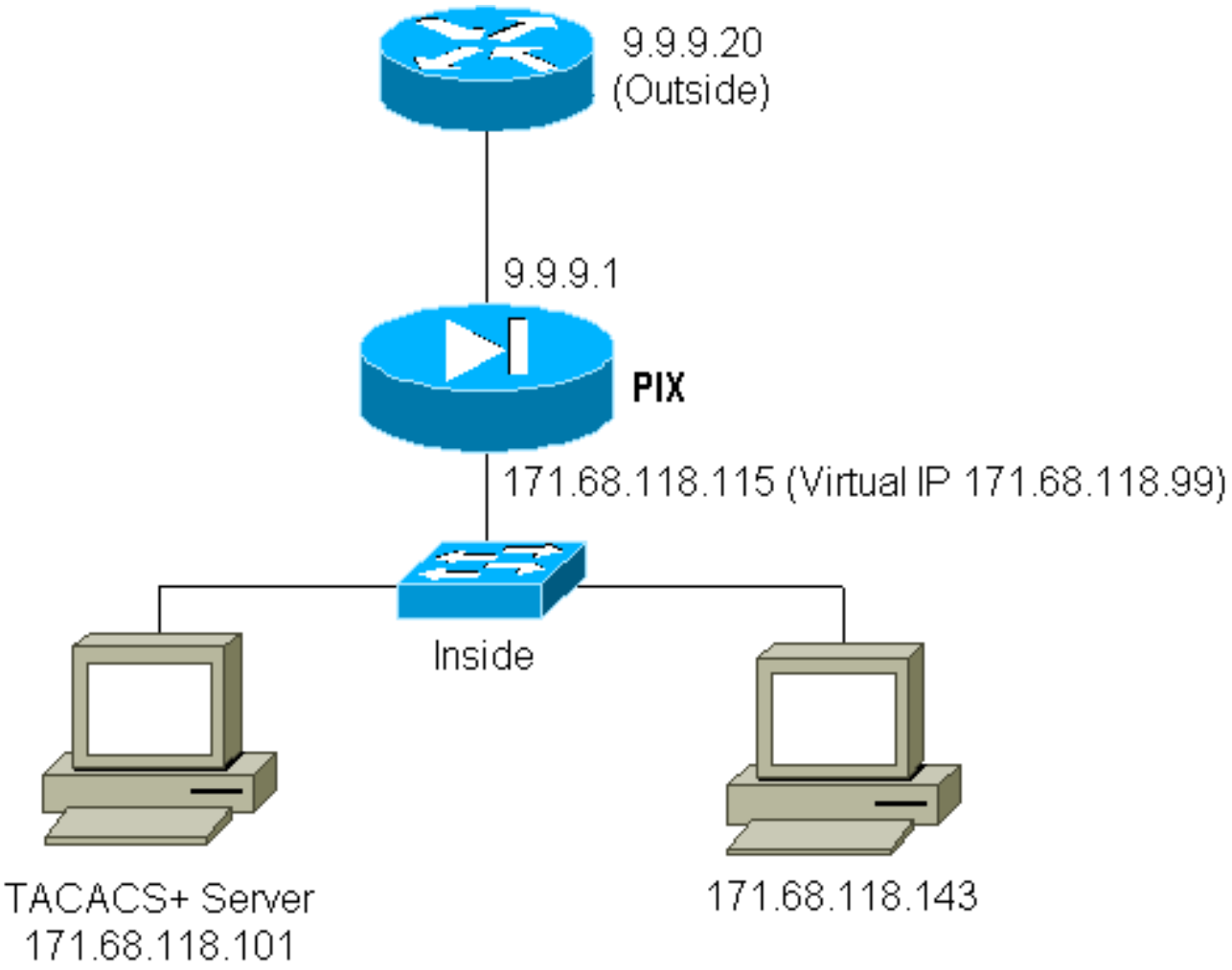
Browser den Benutzernamen und das Kennwort zwischenspeichern.

Um dies zu vermeiden, können Sie virtuelles HTTP implementieren, indem Sie der PIX-Konfiguration eine RFC 1918 -Adresse (d. h. 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.

Beispiel für ausgehenden virtuellen HTTP-Verkehr:



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 Outgoing protocol tacacs+
```

```
aaa-server Outgoing (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 Outgoing
virtual http 171.68.118.99
auth-prompt THIS_IS_PIX_5
```

Virtuelles Telnet

Die Konfiguration des PIX zur Authentifizierung des gesamten ein- und ausgehenden Datenverkehrs ist nicht empfehlenswert, da 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:

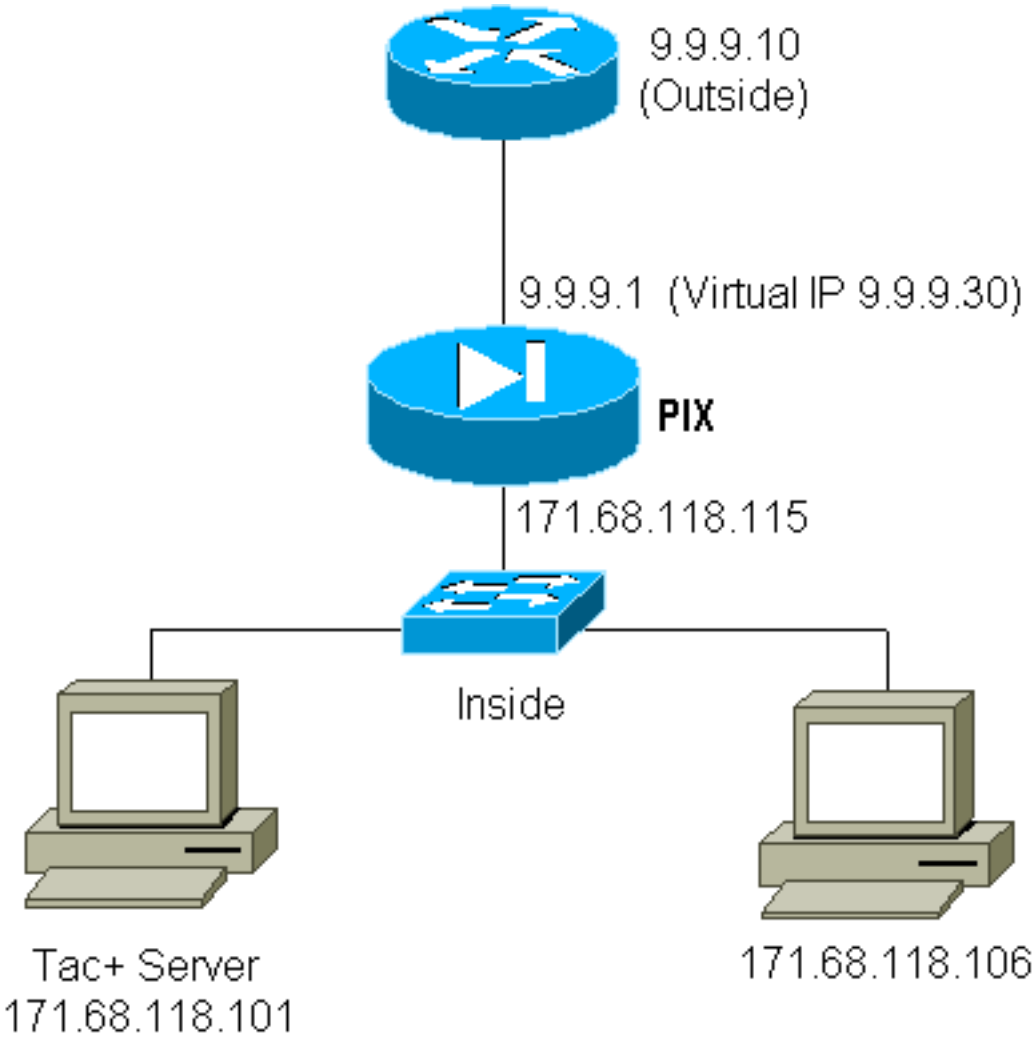
```
109001: Auth start for user '???' from 9.9.9.10/11094 to 171.68.118.106/25
109009: Authorization denied from 171.68.118.106/49 to 9.9.9.10/11094
(not authenticated
```

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 jedoch wirklich eine Art ungewöhnlicher Dienst 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, der an die virtuelle IP gebunden ist.

In unserem Beispiel möchten wir zulassen, dass der TCP-Port 49-Datenverkehr von dem externen Host 9.9.9.10 an den internen Host 171.68.118.106 fließt. Da dieser Datenverkehr nicht wirklich authentifizierbar ist, wird ein virtuelles Telnet eingerichtet.

Virtual Telnet Inbound:



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) 9.9.9.30 171.68.118.106 netmask 255.255.255.255 0 0
 conduit permit tcp host 9.9.9.30 host 9.9.9.10
 aaa-server TACACS+ protocol tacacs+
 aaa-server Incoming protocol tacacs+
 aaa-server Incoming (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 Incoming
 virtual telnet 9.9.9.30
```

TACACS+ Server User Configuration 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 zunächst per Telnet an die Adresse 9.9.9.30 auf dem PIX authentifizieren:

```
pixfirewall# 109001: Auth start for user '???' from 9.9.9.10/11099
 to 171.68.118.106/23
 109011: Authen Session Start: user 'pinecone', sid 13
 109005: Authentication succeeded for user 'pinecone' from
 171.68.118.106/23 to 9.9.9.10/11099
```

Nach erfolgreicher Authentifizierung zeigt der Befehl **show uauth** an, 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
```

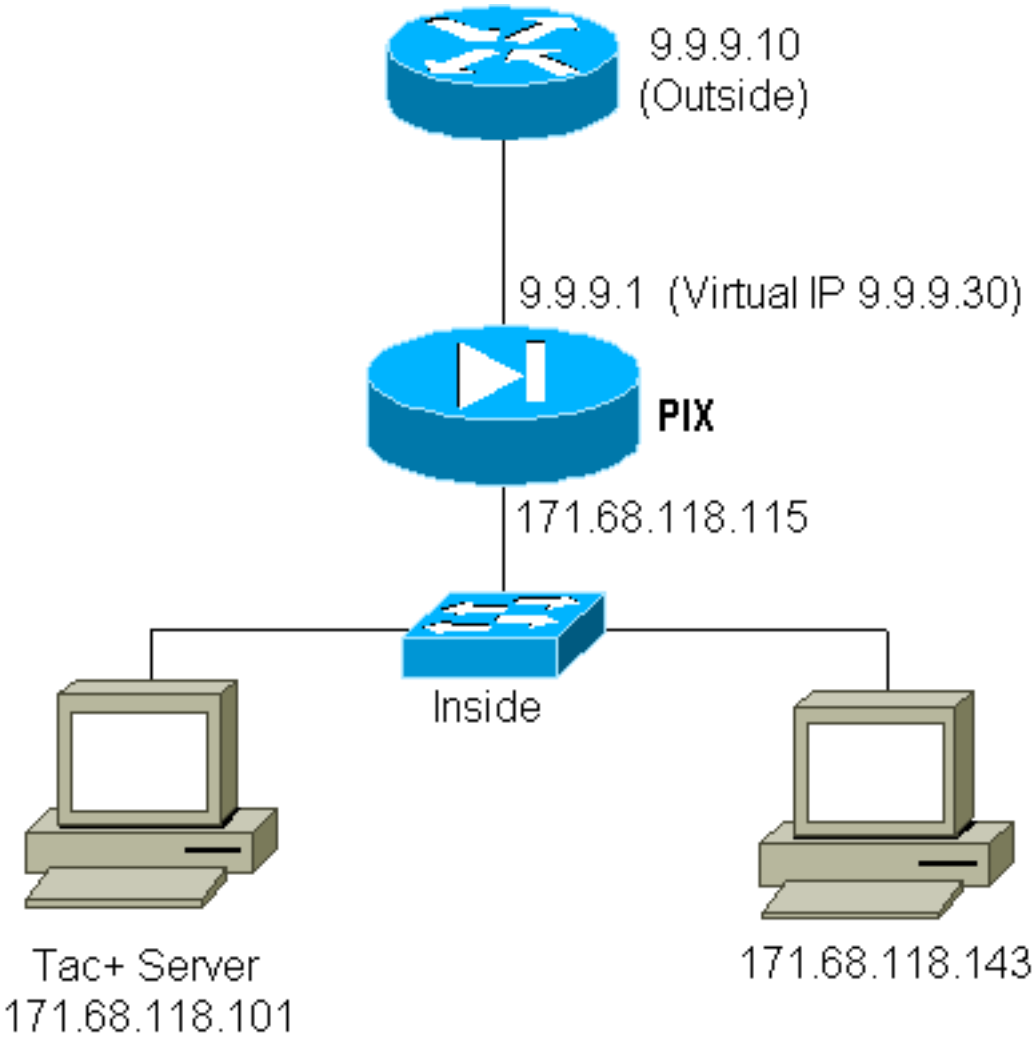
Wenn das Gerät unter 9.9.9.10 TCP/49-Datenverkehr an das Gerät unter 171.68.118.106 senden möchte:

```
pixfirewall# 109001: Auth start for user 'pinecone'
 from 9.9.9.10/11104 to 171.68.118.106/49
 109011: Authen Session Start: user 'pinecone', sid 14
 109007: Authorization permitted for user 'pinecone' from 9.9.9.10/11104
 to 171.68.118.106/49
 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 - ausgehender Datenverkehr:

Da ausgehender Datenverkehr standardmäßig zulässig ist, ist für die Verwendung von ausgehenden virtuellen Telnet-Verbindungen kein statisches Gerät erforderlich. Im folgenden Beispiel wird der interne Benutzer mit der Nummer 171.68.118.143 Telnet zu virtual 9.9.9.30 übertragen und authentifiziert. 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 Configuration 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 Outgoing protocol tacacs+
 aaa-server Outgoing (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 Outgoing
 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 Telnets zur virtuellen Telnet-IP-Adresse wechselt, wird der Befehl **show uauth** angezeigt. Wenn der Benutzer verhindern möchte, dass Datenverkehr nach Abschluss der Sitzung weitergeleitet wird (wenn noch Zeit in der Warteschlange verbleibt), muss er 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. Im folgenden Beispiel war noch Authentifizierung für alle ausgehenden Datenverkehr erforderlich, aber die Autorisierung ist nur 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 Outgoing
aaa authorization tcp/23-49 outbound 0.0.0.0 0.0.0.0 0.0.0.0 0.0.0.0 Outgoing
```

Wenn wir also Telnet von 171.68.118.143 bis 9.9.9.10 verwenden, sind Authentifizierung und Autorisierung erfolgt, weil der Telnet-Port 23 im Bereich von 23 bis 49 liegt. Wenn wir eine HTTP-Sitzung von 171.68.118.143 bis 9.9.9.10 durchführen, müssen wir 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)
```

Zugehörige Informationen

- Produkt-Support für die Cisco PIX Firewall
- Cisco Secure PIX Firewall - Befehlsreferenzen