

802.1x DACL、ユーザごとの ACL、フィルタ ID およびデバイス トラッキングの動作

目次

[概要](#)

[デバイス トラッキング理論](#)

[デバイス トラッキング設定](#)

[テストをトラッキングするデバイス](#)

[バージョン 12.2.33 からのデバッグ、DHCPスヌーピングによってアップデートされる IPデバイス トラッキング](#)

[プローブおよび ARP スヌーピング](#)

[バージョン 12.2.55 のためにトラッキングする IPデバイス-隠しコマンド](#)

[バージョン 12.2.55 のためにトラッキングする IPデバイス-静的な IP 例](#)

[バージョン 15.x のためにトラッキングする IPデバイス](#)

[Cisco IOS XE[®] のためにトラッキングする IPデバイス](#)

[バージョン 12.2.55 のための 802.1X および DACL とトラッキングする IPデバイス](#)

[バージョン 15.x のための 802.1X および DACL とトラッキングする IPデバイス](#)

[特定の ACL項目](#)

[コントロール方向](#)

[バージョン 15.x のための 802.1X およびユーザごとの ACL とトラッキングする IPデバイス](#)

[DACL と比較された場合違い](#)

[バージョン 15.x のための 802.1X およびフィルタid ACL とトラッキングする IPデバイス](#)

[IPデバイス トラッキング-デフォルトおよび最良の方法](#)

[バージョン 15.x のためのインターフェイス ACL 書き換え](#)

[802.1X に使用するデフォルトACL](#)

[モードを開いて下さい](#)

[インターフェイス ACL が必須である時](#)

[4500/6500 の DACL](#)

[802.1X のための MAC アドレス ステータス](#)

[トラブルシューティング](#)

[関連情報](#)

概要

トリガーがホストを追加し、取除くためにであるものが含まれている IPデバイス トラッキング機能がどのように動作するかこの資料に記述されています。また、トラッキングする 802.1X ダウンロード可能 アクセス制御リスト (DACL) のデバイスの影響は説明されます。バージョンとプラットフォーム間の動作変更。

資料の第 2 一部は認証、許可、アカウンティング (AAA) サーバによって戻り、802.1X にセッ

ションを適用される Access Control List (ACL) に焦点を合わせます。 DACL、ユーザごとの ACL およびフィルタid ACL 間の比較は示されます。 また、ACL 書き直しに関する警告およびデフォルトACL は説明されています。

デバイストラッキング理論

デバイストラッキングは次の場合にはエントリを追加します:

- それは DHCPスヌーピングによって New エントリを学びます。
- それはアドレス解決プロトコル (ARP) 要求によって New エントリを学びます (ARPパケットからの送信側 MAC アドレスおよび送信側 IP アドレスを読み込みます)。 機能性は時々 ARP インスペクション、それと呼出されるがこと同じではないですダイナミック ARP 検査 (戴) と。 機能はデフォルトで有効になり、無効である場合もないこと。 それはまた ARP スヌーピングと呼ばれますが、「スヌーピングする」が debug arp 有効になった後デバッグはそれを示しません。 ARP スヌーピングはデフォルトで有効になり、無効または制御されます。

デバイストラッキングは ARP要求のための無応答がないときエントリを削除します (デバイス追跡テーブル、デフォルトで 30 秒毎にの各ホストのためのプローブを送信する)。

デバイストラッキング設定

```
ip dhcp excluded-address 192.168.0.1 192.168.0.240
ip dhcp pool POOL
  network 192.168.0.0 255.255.255.0
!
ip dhcp snooping vlan 1
ip dhcp snooping
ip device tracking
!
interface Vlan1
ip address 192.168.0.2 255.255.255.0
ip route 0.0.0.0 0.0.0.0 10.48.66.1
!
interface FastEthernet0/1
description PC
```

テストをトラッキングするデバイス

```
BSNS-3560-1# show ip dhcp binding
IP address      Client-ID/
                Hardware address
192.168.0.241   0100.5056.994e.a1   Mar 02 1993 02:31 AM   Automatic
```

```
BSNS-3560-1# show ip device tracking all
IP Device Tracking = Enabled
```

```
-----
IP Address      MAC Address      Interface          STATE
-----
192.168.0.241   0050.5699.4ea1   FastEthernet0/1   ACTIVE
```

バージョン 12.2.33 からのデバッグ、DHCPスヌーピングによってアップデートさ

れる IP デバイス トラッキング

DHCP スヌーピングは バインディング テーブルを読み込みます:

```
BSNS-3560-1# show debugging
```

```
DHCP Snooping packet debugging is on
```

```
DHCP Snooping event debugging is on
```

```
DHCP server packet debugging is on.
```

```
DHCP server event debugging is on.
```

```
track:
```

```
IP device-tracking redundancy events debugging is on
```

```
IP device-tracking cache entry Creation debugging is on
```

```
IP device-tracking cache entry Destroy debugging is on
```

```
IP device-tracking cache events debugging is on
```

```
02:30:57: DHCP_SNOOPING: checking expired snoop binding entries
```

```
02:31:12: DHCP_SNOOP(hl_fm_set_if_input): Setting if_input to Fa0/1 for pak. Was V11
```

```
02:31:12: DHCP_SNOOP(hl_fm_set_if_input): Setting if_input to V11 for pak. Was Fa0/1
```

```
02:31:12: DHCP_SNOOP(hl_fm_set_if_input): Setting if_input to Fa0/1 for pak. Was V11
```

```
02:31:12: DHCP_SNOOPING: received new DHCP packet from input interface (FastEthernet0/1)
```

```
02:31:12: DHCP_SNOOPING: process new DHCP packet, message type: DHCPREQUEST, input
```

```
interface: Fa0/1, MAC da: 001f.27e6.cfc0, MAC sa: 0050.5699.4ea1, IP da: 192.168.0.2,
```

```
IP sa: 192.168.0.241, DHCP ciaddr: 192.168.0.241, DHCP yiaddr: 0.0.0.0,
```

```
DHCP siaddr: 0.0.0.0, DHCP giaddr: 0.0.0.0, DHCP chaddr: 0050.5699.4ea1
```

```
02:31:12: DHCP_SNOOPING: add relay information option.
```

```
02:31:12: DHCP_SNOOPING_SW: Encoding opt82 CID in vlan-mod-port format
```

```
02:31:12: DHCP_SNOOPING_SW: Encoding opt82 RID in MAC address format
```

```
02:31:12: DHCP_SNOOPING: binary dump of relay info option, length: 20 data: colon;
```

```
0x52 0x12 0x1 0x6 0x0 0x4 0x0 0x1 0x1 0x3 0x2 0x8 0x0 0x6 0x0 0x1F 0x27 0xE6 0xCF 0x80
```

```
02:31:12: DHCP_SNOOPING_SW: bridge packet get invalid mat entry: 001F.27E6.CFC0,
```

```
packet is flooded to ingress VLAN: (1)
```

```
02:31:12: DHCP_SNOOPING_SW: bridge packet send packet to cpu port: Vlan1.
```

```
02:31:12: DHCPD: DHCPREQUEST received from client 0100.5056.994e.a1.
```

```
02:31:12: DHCPD: Sending DHCPACK to client 0100.5056.994e.a1 (192.168.0.241).
```

```
02:31:12: DHCPD: unicasting BOOTREPLY to client 0050.5699.4ea1 (192.168.0.241).
```

```
02:31:12: DHCP_SNOOPING: received new DHCP packet from input interface (Vlan1)
```

```
02:31:12: DHCP_SNOOPING: process new DHCP packet, message type: DHCPACK, input interface:
```

```
V11, MAC da: 0050.5699.4ea1, MAC sa: 001f.27e6.cfc0, IP da: 192.168.0.241,
```

```
IP sa: 192.168.0.2, DHCP ciaddr: 192.168.0.241, DHCP yiaddr: 192.168.0.241,
```

```
DHCP siaddr: 0.0.0.0, DHCP giaddr: 0.0.0.0, DHCP chaddr: 0050.5699.4ea1
```

```
02:31:12: DHCP_SNOOPING: add binding on port FastEthernet0/1.
```

```
02:31:12: DHCP_SNOOPING: added entry to table (index 189)
```

```
02:31:12: DHCP_SNOOPING: dump binding entry: Mac=00:50:56:99:4E:A1 Ip=192.168.0.241
```

```
Lease=86400 ld Type=dhcp-snooping Vlan=1 If=FastEthernet0/1
```

DHCP バインディングがデータベースに追加された後、デバイス トラッキングのための通知を引き起こします:

```
02:31:12: sw_host_track-ev:host_track_notification: Add event for host 0050.5699.4ea1, 192.168.0.241 on interface FastEthernet0/1
```

```
02:31:12: sw_host_track-ev:Async Add event for host 0050.5699.4ea1, 192.168.0.241
```

```
on interface FastEthernet0/1
```

```
02:31:12: sw_host_track-ev:MSG = 2
```

```
02:31:12: DHCP_SNOOPING_SW no entry found for 0050.5699.4ea1 0.0.0.1 FastEthernet0/1
```

```
02:31:12: DHCP_SNOOPING_SW host tracking not found for update add dynamic
```

```
(192.168.0.241, 0.0.0.0, 0050.5699.4ea1) vlan 1
```

```
02:31:12: DHCP_SNOOPING: direct forward dhcp reply to output port: FastEthernet0/1.
```

```
02:31:12: sw_host_track-ev:Add event: 0050.5699.4ea1, 192.168.0.241, FastEthernet0/1
```

```
02:31:12: sw_host_track-obj_create:0050.5699.4ea1(192.168.0.241) Cache entry created
```

```
02:31:12: sw_host_track-ev:Activating host 0050.5699.4ea1, 192.168.0.241 on
```

```
interface FastEthernet0/1
```

```
02:31:12: sw_host_track-ev:0050.5699.4ea1 Starting cache timer: 30 seconds
```

ARP プローブは 30 秒毎にデフォルトで送信 されます:

```
02:41:12: sw_host_track-ev:0050.5699.4ea1 Stopping cache timer
02:41:12: sw_host_track-ev:0050.5699.4ea1: Send Host probe (0)
02:41:12: sw_host_track-ev:0050.5699.4ea1 Starting cache timer: 30 seconds
02:41:42: sw_host_track-ev:0050.5699.4ea1 Stopping cache timer
02:41:42: sw_host_track-ev:0050.5699.4ea1: Send Host probe (1)
02:41:42: sw_host_track-ev:0050.5699.4ea1 Starting cache timer: 30 seconds
02:42:12: sw_host_track-ev:0050.5699.4ea1 Stopping cache timer
02:42:12: sw_host_track-ev:0050.5699.4ea1: Send Host probe (2)
02:42:12: sw_host_track-ev:0050.5699.4ea1 Starting cache timer: 30 seconds
02:42:42: sw_host_track-ev:0050.5699.4ea1 Stopping cache timer
02:42:42: sw_host_track-obj_destroy:0050.5699.4ea1(192.168.0.241): Cache entry deleted
02:42:42: sw_host_track-ev:0050.5699.4ea1 Stopping cache timer
 3 30.0110700 Cisco_e6:cf:83 vmware_99:4e:a1 ARP 60 who has 192.168.0.241? Tell 0.0.0.0
 4 30.0111260 VMware_99:4e:a1 Cisco_e6:cf:83 ARP 42 192.168.0.241 is at 00:50:56:99:4e:a1
 5 60.0235090 Cisco_e6:cf:83 vmware_99:4e:a1 ARP 60 who has 192.168.0.241? Tell 0.0.0.0
 6 60.0235250 VMware_99:4e:a1 Cisco_e6:cf:83 ARP 42 192.168.0.241 is at 00:50:56:99:4e:a1
 7 90.0230090 Cisco_e6:cf:83 vmware_99:4e:a1 ARP 60 who has 192.168.0.241? Tell 0.0.0.0
 8 90.0230250 VMware_99:4e:a1 Cisco_e6:cf:83 ARP 42 192.168.0.241 is at 00:50:56:99:4e:a1
```

エントリがデバイス追跡テーブルから削除された後、対応する DHCP バインディング エントリはそこにまだあります:

```
BSNS-3560-1#show ip device tracking all
IP Device Tracking = Enabled
```

```
-----
 IP Address      MAC Address      Interface      STATE
-----
```

```
BSNS-3560-1#show ip dhcp binding
```

```
IP address      Client-ID/      Lease expiration      Type
Hardware address
192.168.0.241   0100.5056.994e.a1   Mar 02 1993 03:06 AM   Automatic
```

ARP 応答があるが、エントリをトラッキングするデバイスはとにかく削除されますとき問題があります。不具合はバージョン 12.2.33 にあるためによろしく、バージョン 12.2.55 または 15.x ソフトウェアに現われなかったこと。

またいくつかの違いがあります (スイッチポート無し) L2 ポート (access-port) および L3 ポートと処理するとき。

プローブおよび ARP スヌーピング

ARP スヌーピング 機能とトラッキングするデバイス:

```
BSNS-3560-1#show debugging
```

```
ARP:
 ARP packet debugging is on
Arp Snoop:
 Arp Snooping debugging is on
```

```
03:43:36: sw_host_track-ev:0050.5699.4ea1 Stopping cache timer
03:43:36: sw_host_track-ev:0050.5699.4ea1: Send Host probe (0)
03:43:36: IP ARP: sent req src 0.0.0.0 001f.27e6.cf83,
          dst 192.168.0.241 0050.5699.4ea1 FastEthernet0/1
03:43:36: sw_host_track-ev:0050.5699.4ea1 Starting cache timer: 30 seconds
03:43:36: IP ARP: rcvd rep src 192.168.0.241 0050.5699.4ea1, dst 0.0.0.0 Vlan1
```

バージョン 12.2.55 のためにトラッキングする IP デバイス-隠しコマンド

バージョン 12.2 に関してはそれをアクティブにするために隠しコマンドを使用する必要がある
かもしれません:

```
BSNS-3560-1#show ip device tracking all
```

```
IP Device Tracking = Enabled
IP Device Tracking Probe Count = 2
IP Device Tracking Probe Interval = 30
IP Device Tracking Probe Delay Interval = 0
```

IP Address	MAC Address	Vlan	Interface	STATE
192.168.0.244	0050.5699.4ea1	55	FastEthernet0/1	ACTIVE

```
Total number interfaces enabled: 1
```

```
Enabled interfaces:
```

```
 Fa0/1
```

```
BSNS-3560-1#ip device tracking interface fa0/48
```

```
BSNS-3560-1#show ip device tracking all
```

```
IP Device Tracking = Enabled
IP Device Tracking Probe Count = 2
IP Device Tracking Probe Interval = 30
IP Device Tracking Probe Delay Interval = 0
```

IP Address	MAC Address	Vlan	Interface	STATE
10.48.67.87	000c.2978.825d	1006	FastEthernet0/48	ACTIVE
10.48.67.31	020a.dada.dada	1006	FastEthernet0/48	ACTIVE
10.48.66.245	acf2.c5ed.8171	1006	FastEthernet0/48	ACTIVE
192.168.0.244	0050.5699.4ea1	55	FastEthernet0/1	ACTIVE
10.48.66.193	000c.2997.4ca1	1006	FastEthernet0/48	ACTIVE
10.48.66.186	0050.5699.3431	1006	FastEthernet0/48	ACTIVE

```
Total number interfaces enabled: 2
```

```
Enabled interfaces:
```

```
 Fa0/1, Fa0/48
```

バージョン 12.2.55 のためにトラッキングする IP デバイス-静的な IP 例

この例では、PC は静的 IP アドレスで設定されました。デバッグは ARP 応答 (MSG=2) があつた後、エントリをトラッキングするデバイスは更新済であることを示します。

```
01:03:16: sw_host_track-ev:0050.5699.4ea1 Stopping cache timer
01:03:16: sw_host_track-ev:0050.5699.4ea1: Send Host probe (0)
01:03:16: sw_host_track-ev:0050.5699.4ea1 Starting cache timer: 30 seconds
01:03:16: sw_host_track-ev:host_track_notification: Add event for host 0050.5699.4ea1,
192.168.0.241 on interface FastEthernet0/1, vlan 1
01:03:16: sw_host_track-ev:Async Add event for host 0050.5699.4ea1, 192.168.0.241
on interface FastEthernet0/1
01:03:16: sw_host_track-ev:MSG = 2
01:03:16: sw_host_track-ev:Add event: 0050.5699.4ea1, 192.168.0.241, FastEthernet0/1
01:03:16: sw_host_track-ev:0050.5699.4ea1: Cache entry refreshed
01:03:16: sw_host_track-ev:Activating host 0050.5699.4ea1, 192.168.0.241 on
interface FastEthernet0/1
01:03:16: sw_host_track-ev:0050.5699.4ea1 Starting cache timer: 30 seconds
```

PC からのそう各 ARP 要求はデバイス追跡テーブルをアップデートします (ARP パケットからの送信側 MAC アドレスおよび送信側 IP アドレス)。

バージョン 15.x のためにトラッキングする IP デバイス

いくつかの 802.1X のための DACL のような機能が LAN ライト バージョンでサポートされないことを覚えておくことは重要です (用心して下さい- Cisco Feature Navigator は正しい情報を常に示しません)。

バージョン 12.2 からの隠しコマンドは実行することができまじたり効果をもたらしません。ソフトウェア バージョン 15.x では、トラッキングする IP デバイスは有効になる 802.1X があるインターフェイスのためにだけ (IPDT) デフォルトで有効になります:

```
bsns-3750-5#show ip device tracking all
```

```
IP Device Tracking = Enabled
IP Device Tracking Probe Count = 3
IP Device Tracking Probe Interval = 30
IP Device Tracking Probe Delay Interval = 0
```

```
-----
IP Address      MAC Address    Vlan  Interface          STATE
-----
192.168.10.12   0007.5032.6941 100   GigabitEthernet1/0/1  ACTIVE
192.168.2.200   000c.29d7.0617 1     GigabitEthernet1/0/1  ACTIVE
```

```
Total number interfaces enabled: 2
Enabled interfaces:
  Gi1/0/1, Gi1/0/2
```

```
bsns-3750-5#show run int g1/0/3
```

```
Building configuration...
```

```
Current configuration : 38 bytes
!
interface GigabitEthernet1/0/3
```

```
bsns-3750-5(config)#int g1/0/3
```

```
bsns-3750-5(config-if)#switchport mode access
bsns-3750-5(config-if)#authentication port-control auto
bsns-3750-5(config-if)#do show ip device tracking all
IP Device Tracking = Enabled
IP Device Tracking Probe Count = 3
IP Device Tracking Probe Interval = 30
IP Device Tracking Probe Delay Interval = 0
```

```
-----
IP Address      MAC Address    Vlan  Interface          STATE
-----
192.168.10.12   0007.5032.6941 100   GigabitEthernet1/0/1  ACTIVE
192.168.2.200   000c.29d7.0617 1     GigabitEthernet1/0/1  ACTIVE
```

```
Total number interfaces enabled: 3
Enabled interfaces:
  Gi1/0/1, Gi1/0/2, Gi1/0/3
```

ポートからの 802.1X 設定の削除の後で、IPDT はまたそのポートから取除かれます。ポートステータスは「DOWN」であるかもしれませんが従って「switchport mode access」および「そのポートでアクティブになる IP デバイス トラッキングがあるために authentication ポート コントロール 自動」は持っていることは必要です。10 への最大インターフェイスデバイス制限は設定されま

```
bsns-3750-5(config-if)#ip device tracking maximum ?
```

```
<1-10> Maximum devices
```

Cisco IOS XE[®] のためにトラッキングする IP デバイス

再度 Cisco IOS バージョン 15.x と比較されてと、動作 on Cisco IOS XE 3.3 は変更しました。バ

ージョン 12.2 からの隠しコマンドは廃止ですが、今このエラーは返されます:

```
3850-1# no ip device tracking int g1/0/48
```

```
% Command accepted but obsolete, unreleased or unsupported; see documentation.
```

Cisco IOS XE では、デバイストラッキングはすべてのインターフェイス (設定される 802.1X が
ない) のために物アクティブになります:

```
3850-1#show ip device tracking all
```

```
Global IP Device Tracking for clients = Enabled
```

```
Global IP Device Tracking Probe Count = 3
```

```
Global IP Device Tracking Probe Interval = 30
```

```
Global IP Device Tracking Probe Delay Interval = 0
```

```
-----  
IP Address      MAC Address      Vlan  Interface      Probe-Timeout  
State          Source  
-----  
10.48.39.29     000c.29bd.3cfa  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP  
10.48.39.28     0016.9dca.e4a7  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP  
10.48.76.117    0021.a0ff.5540  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP  
10.48.39.21     00c0.9f87.7471  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP  
10.48.39.16     0050.5699.1093  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP  
10.76.191.247   0024.9769.58cf  20    GigabitEthernet1/0/48  30  
ACTIVE        ARP  
192.168.99.4    d48c.b52f.4a1e  99    GigabitEthernet1/0/12  30  
INACTIVE     ARP  
10.48.39.13     000c.296e.8dbc  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP  
10.48.39.15     0050.5699.128d  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP  
10.48.39.9      0012.da20.8c00  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP  
10.48.39.8      6c20.560e.1b64  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP  
10.48.39.11     000c.29e9.db25  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP  
10.48.39.5      0014.f15f.f7ca  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP  
10.48.39.4      000c.2972.57bc  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP  
10.48.39.7      5475.d029.74cf  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP  
10.48.76.108    001c.58de.9340  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP  
10.48.39.1      0006.f62a.c4a3  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP  
10.48.39.3      0050.5699.1bee  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP  
10.48.76.84     0015.58c5.e8b7  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP  
10.48.39.56     0015.fa13.9a40  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP  
10.48.39.59     0050.5699.1bf4  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP  
10.48.39.58     000c.2957.c7ad  1     GigabitEthernet1/0/48  30  
ACTIVE        ARP
```

```
Total number interfaces enabled: 57
```

```
Enabled interfaces:
Gi1/0/1, Gi1/0/2, Gi1/0/3, Gi1/0/4, Gi1/0/5, Gi1/0/6, Gi1/0/7,
Gi1/0/8, Gi1/0/9, Gi1/0/10, Gi1/0/11, Gi1/0/12, Gi1/0/13, Gi1/0/14,
Gi1/0/15, Gi1/0/16, Gi1/0/17, Gi1/0/18, Gi1/0/19, Gi1/0/20, Gi1/0/21,
Gi1/0/22, Gi1/0/23, Gi1/0/24, Gi1/0/25, Gi1/0/26, Gi1/0/27, Gi1/0/28,
Gi1/0/29, Gi1/0/30, Gi1/0/31, Gi1/0/32, Gi1/0/33, Gi1/0/34, Gi1/0/35,
Gi1/0/36, Gi1/0/37, Gi1/0/38, Gi1/0/39, Gi1/0/40, Gi1/0/41, Gi1/0/42,
Gi1/0/43, Gi1/0/44, Gi1/0/45, Gi1/0/46, Gi1/0/47, Gi1/0/48, Gi1/1/1,
Gi1/1/2, Gi1/1/3, Gi1/1/4, Te1/1/1, Te1/1/2, Te1/1/3, Te1/1/4
3850-1#
```

```
3850-1#sh run int g1/0/48
Building configuration...
```

```
Current configuration : 39 bytes
!
interface GigabitEthernet1/0/48
end
```

```
3850-1(config-if)#ip device tracking maximum ?
<0-65535> Maximum devices (0 means disabled)
```

また、ポートに対して最大エントリのための制限がありません（0は無効を意味します）。

バージョン 12.2.55 のための 802.1X および DACL とトラッキングする IP デバイス

802.1X が DACL で設定される場合デバイスの IP アドレスを一杯にするために、エントリをトラッキングするデバイスは使用されます。この例は構成された IP のために静的にデバイストラッキングにはたらくことを示したものです:

```
BSNS-3560-1#show ip device tracking all
IP Device Tracking = Enabled
IP Device Tracking Probe Count = 2
IP Device Tracking Probe Interval = 30
IP Device Tracking Probe Delay Interval = 0
```

```
-----
IP Address      MAC Address    Vlan  Interface          STATE
-----
192.168.0.244   0050.5699.4ea1  2     FastEthernet0/1    ACTIVE
```

```
Total number interfaces enabled: 1
Enabled interfaces:
Fa0/1
```

これは「割り当て icmp と構築される 802.1X セッションあらゆるあらゆる」DACL です:

```
BSNS-3560-1# sh authentication sessions interface fa0/1
  Interface: FastEthernet0/1
  MAC Address: 0050.5699.4ea1
  IP Address: 192.168.0.244
  User-Name: cisco
  Status: Authz Success
  Domain: DATA
  Security Policy: Should Secure
  Security Status: Unsecure
  Oper host mode: single-host
  Oper control dir: both
  Authorized By: Authentication Server
  Vlan Policy: 2
  ACS ACL: xACSACLx-IP-DACL-516c2694
  Session timeout: N/A
  Idle timeout: N/A
```



```
Common Session ID: 0A3042A900000008008900C5
Acct Session ID: 0x0000000D
Handle: 0x19000008
```

Runnable methods list:

```
Method State
dot1x Authc Success BSNS-3560-1#show epm session summary
```

EPM Session Information

```
-----
Total sessions seen so far : 1
Total active sessions      : 1
```

```
Interface          IP Address          MAC Address          Audit Session Id:
-----
FastEthernet0/1    192.168.0.244      0050.5699.4ea1      0A3042A900000008008900C5
```

これは応用 ACL を示します:

```
BSNS-3560-1#show ip access-lists
Extended IP access list Auth-Default-ACL
 10 permit udp any range bootps 65347 any range bootpc 65348
 20 permit udp any any range bootps 65347
 30 deny ip any any (8 matches)
```

```
Extended IP access list xACSACLx-IP-DACL-516c2694 (per-user)
 10 permit icmp any any (6 matches)
```

また、fa0/1 インターフェイスの ACL は同じです:

```
BSNS-3560-1#show ip access-lists interface fa0/1
 permit icmp any any
```

デフォルトが dot1x ACL であるのに:

```
BSNS-3560-1#show ip interface fa0/1
FastEthernet0/1 is up, line protocol is up
 Inbound access list is Auth-Default-ACL
```

それは ACL が 192.168.0.244 として「どれでも」使用することができるように期待されるかもしれませんが、auth プロキシのための、しかし 802.1X DACL ソース「どれでも」のためのこのような作業は PC の検出する IP に変更されないこと。

auth プロキシに関しては、ACS からの 1 つのオリジナル ACL はキャッシュされ、show ip access-list コマンドおよび特定の (特定の IP とユーザごと) ACL と示されていて提示 IP アクセスリスト インターフェイス fa0/1 コマンドでインターフェイスで適用されます。ただし、auth-proxy はデバイス IP トラッキングを使用しません。

IP アドレスが正しく検出する場合はどうしたらいいのですか。デバイストラッキングの後で無効です:

```
BSNS-3560-1#show authentication sessions interface fa0/1
Interface: FastEthernet0/1
MAC Address: 0050.5699.4ea1
IP Address: Unknown
User-Name: cisco
Status: Authz Success
Domain: DATA
Security Policy: Should Secure
Security Status: Unsecure
Oper host mode: single-host
Oper control dir: both
Authorized By: Authentication Server
Vlan Policy: 2
ACS ACL: xACSACLx-IP-DACL-516c2694
```

```
Session timeout: N/A
Idle timeout: N/A
Common Session ID: 0A3042A90000000000000000C775
Acct Session ID: 0x00000001
Handle: 0xB0000000
```

Runnable methods list:

```
Method State
dot1x Authc Success
```

そう IP アドレスはそれから接続されませんが、DACL はまだ適用されます:

```
BSNS-3560-1#show ip access-lists
```

```
Extended IP access list Auth-Default-ACL
```

```
10 permit udp any range bootps 65347 any range bootpc 65348
20 permit udp any any range bootps 65347
30 deny ip any any (4 matches)
```

```
Extended IP access list xACSACLx-IP-DACL-516c2694 (per-user)
```

```
10 permit icmp any any
```

このシナリオでは、802.1X のためにトラッキングするデバイスが必要となりません。唯一の違いはクライアントの IP アドレスを知っていることが RADIUS access-request に upfront 使用することができます。アトリビュート 8 の後で接続されます:

```
radius-server attribute 8 include-in-access-req
```

それは Access-Request にあり、ACS で粒状承認規則を作成することは可能性のあるです:

```
00:17:44: RADIUS(00000001): Send Access-Request to 10.48.66.185:1645 id 1645/27, len 257
00:17:44: RADIUS: authenticator F8 17 06 CE C1 85 E8 E8 - CB 5B 57 96 6C 07 CE CA
00:17:44: RADIUS: User-Name [1] 7 "cisco"
00:17:44: RADIUS: Service-Type [6] 6 Framed [2]
00:17:44: RADIUS: Framed-IP-Address [8] 6 192.168.0.244
```

その TrustSec にまた必要とします SGT バインディングに IP のためにトラッキングする IP デバイスを留意して下さい。

バージョン 15.x のための 802.1X および DACL とトラッキングする IP デバイス

DACL のバージョン 15.x とバージョン 12.2.55 の違いとは何か。ソフトウェア Version 15.x では、それは auth-proxy のためにと同じをはたらかせます。一般的な ACL は show ip access-list コマンドが (AAA からのキャッシュされた応答) 入力されるがとき、提示 IP アクセスリスト インターフェイス fa0/1 コマンドがホストのソース IP アドレスと、src 「」取替えられた後見られる場合があります (トラッキングする IP デバイスによって知られている)。

これは 1 つのポート (g1/0/1) の電話および PC のための例、3750X のソフトウェア バージョン 15.0.2SE2 です:

```
bsns-3750-5#sh authentication sessions interface g1/0/1
```

```
Interface: GigabitEthernet1/0/1
MAC Address: 0007.5032.6941
IP Address: 192.168.10.12
User-Name: 00-07-50-32-69-41
Status: Authz Success
Domain: VOICE
Security Policy: Should Secure
Security Status: Unsecure
Oper host mode: multi-auth
Oper control dir: both
Authorized By: Authentication Server
Vlan Policy: 100
```

```
ACS ACL: xACSACLx-IP-PERMIT_ALL_TRAFFIC-51134bb2
Session timeout: N/A
Idle timeout: N/A
Common Session ID: C0A80001000001012B680D23
Acct Session ID: 0x0000017B
Handle: 0x99000102
```

Runnable methods list:

```
Method State
dot1x Failed over
mab Authc Success
```

```
-----
Interface: GigabitEthernet1/0/1
MAC Address: 0050.5699.4ea1
IP Address: 192.168.2.200
User-Name: cisco
Status: Authz Success
Domain: DATA
Security Policy: Should Secure
Security Status: Unsecure
Oper host mode: multi-auth
Oper control dir: both
Authorized By: Authentication Server
Vlan Policy: 20
ACS ACL: xACSACLx-IP-PERMIT_ALL_TRAFFIC-51134bb2
Session timeout: N/A
Idle timeout: N/A
Common Session ID: C0A80001000001BD336EC4D6
Acct Session ID: 0x000002F9
Handle: 0xF80001BE
```

Runnable methods list:

```
Method State
dot1x Authc Success
mab Not run
```

電話は MAC 認証 バイパス (MAB) によって PC は dot1x を使用するが、認証されます。 電話および PC は両方同じ ACL を使用します:

```
bsns-3750-5#show ip access-lists xACSACLx-IP-PERMIT_ALL_TRAFFIC-51134bb2
Extended IP access list xACSACLx-IP-PERMIT_ALL_TRAFFIC-51134bb2 (per-user)
 10 permit ip any any
```

ただし、インターフェイス レベルで確認されたときソースはデバイスの IP アドレスと取替えられました。 変更するおよびそれはいつでも発生する場合がありますトリガーをトラッキングする IP デバイス (ACL の認証 セッションおよびダウンロードより大いにあとで):

```
bsns-3750-5#show ip access-lists interface g1/0/1
 permit ip host 192.168.2.200 any (5 matches)
 permit ip host 192.168.10.12 any
```

MAC アドレスは両方ともスタティックとしてマークする必要があります:

```
bsns-3750-5#sh mac address-table interface g1/0/1
Mac Address Table
```

```
-----
Vlan Mac Address Type Ports
----
20 0050.5699.4ea1 STATIC Gi1/0/1
100 0007.5032.6941 STATIC Gi1/0/1
```

特定の ACL項目

ソースはいつ DACL の「どれでも」ホスト IP アドレスと取り替えられますか。同じポート (2人の要求元) に少なくとも 2 セッションがある時だけ。

セッションが 1 だけあるときソースを「」取り替える必要がありません。問題はマルチセッションがある、すべてのために IP デバイストラッキングはホストの IP アドレスを知っていませんとき現われるかもしれないし。そのシナリオでそれはまだいくつかのエントリのため「どれでも」です。

その動作はいくつかのプラットフォームで異なっています。たとえば、バージョン 15.0(2)EX との 2960X で ACL はポートに対してちょうど 1 認証セッションがある時でさえ特定常にです。ただし、3560X および 3750X バージョン 15.0(2)SE のために、その ACL 細目を作る少なくとも 2 セッションがある必要があります。

コントロール方向

デフォルトで、コントロール方向は型両方です:

```
bsns-3750-5(config)#int g1/0/1
bsns-3750-5(config-if)#authentication control-direction ?
  both   Control traffic in BOTH directions
  in     Control inbound traffic only
```

```
bsns-3750-5(config-if)#authentication control-direction both
```

それはサブリカントが認証される前に、トラフィックはポートで送受信することができないことを意味します。のためにモード「で」、トラフィックはポートからサブリカントに、ないサブリカントからポートに送信されたかもしれません (LAN 機能の航跡に役立つ可能性があります)。

それにもかかわらず、スイッチは「」方向の「の ACL をちょうど適用します。どのによってモードが使用されるか重要ではありません。

```
bsns-3750-5#sh ip access-lists interface g1/0/1 out
bsns-3750-5#sh ip access-lists interface g1/0/1 in
  permit ip host 192.168.2.200 any
  permit ip host 192.168.10.12 any
```

それは認証が ACL (方向) のポートにトラフィックに適用した、後ことを基本的に意味しますポート (方向) からすべてのトラフィックが割り当てられます。

バージョン 15.x のための 802.1X およびユーザごとの ACL とトラッキングする IP デバイス

cisco-av-pair で「IP 渡されるユーザごとの ACL を使用することもまた可能性のあるです: inacl」および「IP: outacl」。

この設定例は以前のコンフィギュレーションに類似したですが、今回電話は DACL および PC 使用ユーザごとの ACL を使用します。PC のための ISE プロファイルは次のとおりです:

▼ Attributes Details

```
Access Type = ACCESS_ACCEPT
Tunnel-Private-Group-ID = 1:20
Tunnel-Type=1:13
Tunnel-Medium-Type=1:6
cisco-av-pair = ip:inacl#1=permit icmp any any log
cisco-av-pair = ip:outacl#1=permit icmp any any
```

電話にまだ適用される DACL があります:

```
bsns-3750-5#show authentication sessions interface g1/0/1
    Interface: GigabitEthernet1/0/1
    MAC Address: 0007.5032.6941
    IP Address: 192.168.10.12
    User-Name: 00-07-50-32-69-41
    Status: Authz Success
    Domain: VOICE
    Security Policy: Should Secure
    Security Status: Unsecure
    Oper host mode: multi-auth
    Oper control dir: both
    Authorized By: Authentication Server
    Vlan Policy: 100
    ACS ACL: xACSACLx-IP-PERMIT_ALL_TRAFFIC-51134bb2
    Session timeout: N/A
    Idle timeout: N/A
    Common Session ID: C0A8000100000568431143D8
    Acct Session ID: 0x000006D2
    Handle: 0x84000569
```

Runnable methods list:

Method	State
dot1x	Failed over
mab	Authc Success

```
bsns-3750-5#sh ip access-lists xACSACLx-IP-PERMIT_ALL_TRAFFIC-51134bb2
Extended IP access list xACSACLx-IP-PERMIT_ALL_TRAFFIC-51134bb2 (per-user)
 10 permit ip any any
```

ただし、同じポートの PC はユーザごとの ACL を使用します:

```
Interface: GigabitEthernet1/0/1
    MAC Address: 0050.5699.4ea1
    IP Address: 192.168.2.200
    User-Name: cisco
    Status: Authz Success
    Domain: DATA
    Security Policy: Should Secure
    Security Status: Unsecure
    Oper host mode: multi-auth
    Oper control dir: both
    Authorized By: Authentication Server
    Vlan Policy: 20
    Per-User ACL: permit icmp any any log
    Session timeout: N/A
    Idle timeout: N/A
    Common Session ID: C0A80001000005674311400B
    Acct Session ID: 0x000006D1
    Handle: 0x9D000568
```

それが gig1/0/1 ポートでどのようにマージされるか確認するため:

```
bsns-3750-5#show ip access-lists interface g1/0/1
  permit icmp host 192.168.2.200 any log
  permit ip host 192.168.10.12 any
```

最初のエントリはユーザごとの ACL から (注意します log キーワードに) 奪取され、第 2 エントリは DACL から奪取されます。両方は特定の IP アドレスのためにトラッキングする IP デバイスによって書き換えられます。

ユーザごとの ACL はデバッグ `epm` とすべてのコマンド確認できます:

```
Apr 12 02:30:13.489: EPM_SESS_EVENT:IP Per-User ACE: permit icmp any any log received
Apr 12 02:30:13.489: EPM_SESS_EVENT:Recieved string GigabitEthernet1/0/1#IP#7844C6C
Apr 12 02:30:13.489: EPM_SESS_EVENT:Add ACE [permit icmp any any log] to ACL
[GigabitEthernet1/0/1#IP#7844C6C]
Apr 12 02:30:13.497: EPM_SESS_EVENT:Executed [ip access-list extended
GigabitEthernet1/0/1#IP#7844C6C] command through parse_cmd. Result= 0
Apr 12 02:30:13.497: EPM_SESS_EVENT:Executed [permit icmp any any log]
command through parse_cmd. Result= 0
Apr 12 02:30:13.497: EPM_SESS_EVENT:Executed [end] command through
parse_cmd. Result= 0
Apr 12 02:30:13.497: EPM_SESS_EVENT:Notifying PD regarding Policy (NAMED ACL)
application on the interface GigabitEthernet1/0/1
```

そしてまた `show ip access-lists` コマンドによって:

```
bsns-3750-5#show ip access-lists
Extended IP access list GigabitEthernet1/0/1#IP#7844C6C (per-user)
  10 permit icmp any any log
```

IP について何: `outacl` アトリビュートか。それはバージョン 15.x で完全に省略されます。アトリビュートは受け取られましたが、スイッチはプロセス適用するために帰因させる。

DACL と比較された場合違い

Cisco バグ ID [CSCut25702](#) に記載のとおり、ユーザごとの ACL は DACL と動作が異なります。ちょうど 1 つのエントリ (「permit ip any any」) およびポートに接続される有効になる IP デバイストラッキングしないで 1 サプリカントの DACL は正しくはたらくことができます。「どの」引数でも代わりにならないし、すべてのトラフィックが割り当てられます。ただし IP デバイストラッキングを有効にしてもらうように、なぜならユーザごとの ACL それは必須です。それは無効で、ちょうど「permit ip any any」エントリおよび 1 サプリカントがある場合、すべてのトラフィックはブロックされます。

バージョン 15.x のための 802.1X およびフィルタid ACL とトラッキングする IP デバイス

また、IETF アトリビュート フィルタid [11] は使用することができます。AAA サーバはスイッチでローカルで定義する必要がある ACL 名前を戻します。ISE プロファイルはこのようになる可能性があります:

▼ Common Tasks

DACL Name

VLAN Tag ID 1 Edit Tag ID/Name 20

Voice Domain Permission

Web Authentication

Auto Smart Port

Filter-ID .in

specify 方向を必要とすることに注目して下さい (でまたは)。 それのためにアトリビュートを手動で追加することは必要です:

▼ Advanced Attributes Settings

=

それからデバッグは示します:

```
debug epm all
```

```
Apr 12 23:41:05.170: EPM_SESS_EVENT:Filter-Id : Filter-ACL received
```

```
Apr 12 23:41:05.170: EPM_SESS_EVENT:Notifying PD regarding Policy (NAMED ACL)  
application on the interface GigabitEthernet1/0/1
```

その ACL はまた認証された セッションのために示されます:

```
bsns-3750-5#show authentication sessions interface g1/0/1
```

```
Interface: GigabitEthernet1/0/1  
MAC Address: 0050.5699.4ea1  
IP Address: 192.168.2.200  
User-Name: cisco  
Status: Authz Success  
Domain: DATA  
Security Policy: Should Secure  
Security Status: Unsecure  
Oper host mode: multi-auth  
Oper control dir: both  
Authorized By: Authentication Server  
Vlan Policy: 20  
Filter-Id: Filter-ACL  
Session timeout: N/A  
Idle timeout: N/A  
Common Session ID: C0A800010000059E47B77481  
Acct Session ID: 0x00000733  
Handle: 0x5E00059F
```

Runnable methods list:

```
Method State  
dot1x Authc Success
```

```
mab      Not run
```

そして、ACL がインターフェイスに binded ように:

```
bsns-3750-5#show ip access-lists interface g1/0/1
  permit icmp host 192.168.2.200 any log
  permit tcp host 192.168.2.200 any log
```

この ACL が同じインターフェイスの ACL の他の型とマージすることができることに注目して下さい。たとえば、別のサブリカントを ISE から DACL を得る同じスイッチポートで持っています: 「permit ip any any」見る可能性がある:

```
bsns-3750-5#show ip access-lists interface g1/0/1
  permit icmp host 192.168.2.200 any log
  permit tcp host 192.168.2.200 any log
  permit ip host 192.168.10.12 any
```

IP デバイストラッキングが各ソース (サブリカント) におけるソース IP を書き換えることに注目して下さい。

何「」フィルタリストについてか。再度 (ユーザごとの ACL として)、それはスイッチによって使用されません。

IP デバイストラッキング-デフォルトおよび最良の方法

先のリリースどの機能でもそれ IPDT を使用できる前に 15.2(1)E よりに関しては、この CLI コマンドでグローバルに最初に有効にされる必要があります:

```
(config)#ip device tracking
```

リリース 15.2(1)E およびそれ以降に関しては、コマンドをトラッキングする IP デバイスはもう必要とされません。IPDT はそれに頼る機能がそれを有効にするときだけ有効になります。機能が IPDT を有効にしない場合、IPDT は無効です。コマンドを「トラッキングする」IP デバイスは効果をもたらしません。特定の機能に IPDT を有効または無効にする制御があります。

IPDT を有効にするとき、「Duplicate IP Address」問題について覚えなければなりません。[「Duplicate IP Address 0.0.0.0」エラーメッセージ](#)を詳細については[解決するために参照](#)して下さい。

トランクポートの IPDT をディセーブルにすることを推奨します:

```
(config-if)# no ip device tracking
```

より遅い Cisco IOS で、それは別のコマンドです:

```
(config-if)#ip device tracking maximum 0
```

"Duplicate IP Address"問題を避けることをアクセスポートおよび遅延 ARP プロブの IPDT が可能にすることを推奨します:

```
(config-if)#ip device tracking probe delay 10
```

バージョン 15.x のためのインターフェイス ACL 書き換え

インターフェイス ACL に関しては、それは認証の前にはたります:

```
interface GigabitEthernet1/0/2
  description windows7
  switchport mode access
```



```
ip access-group test1 in
authentication order mab dot1x
authentication port-control auto
mab
dot1x pae authenticator
end
```

```
bsns-3750-5#show ip access-lists test1
Extended IP access list test1
 10 permit tcp any any log-input
```

ただし、認証は成功した後 AAAサーバ (DACL ならから戻る ACL によって (上書きする)、IP 重要ではありません書き換えられます: inacl、か filterid)。

その ACL (test1) はブロック普通許可される)、トラフィック 認証がもう重要ではなかった後 (開いたモードでできますが。ACL が AAAサーバから戻らない時でさえ、インターフェイス ACL は上書きされ、フルアクセスは提供されます。それは少し ACL はまだインターフェイスレベルで binded ことを Ternary Content Addressable Memory (TCAM) が示すので誤解していません。3750X のバージョン 15.2.2 からの例はここにあります:

```
bsns-3750-6#show platform acl portlabels interface g1/0/2
```

```
Port based ACL: (asic 1)
-----
Input Label: 5      Op Select Index: 255
Interface(s): Gi1/0/2
Access Group: test1, 4 VMRs
Ip Portal: 0 VMRs
IP Source Guard: 0 VMRs
LPIP: 0 VMRs
AUTH: 0 VMRs
C3PLACL: 0 VMRs
MAC Access Group: (none), 0 VMRs
```

その情報はインターフェイスレベルのためにだけ、ない水平なセッションのために有効です。もう少しの情報は (混合された ACL を示します) から推論することができます:

```
bsns-3750-6#show ip access-lists interface g1/0/2
 permit ip host 192.168.1.203 any
Extended IP access list test1
 10 permit icmp host 2.2.2.2 host 1.1.1.1
```

最初のエントリは認証の成功のために「permit ip any any」DACL として戻ります作成され、(「」デバイス追跡テーブルからのエントリによって取り替えられます)。第2エントリはインターフェイス ACL の結果で、すべての新しい認証に適用します (許可の前に)。

残念ながら、(再度依存したプラットフォーム) ACL は両方とも連結されます。それは 3750X のバージョン 15.2.2 で起こります。それは承認されたセッションのためにそれを、両方適用します意味します。最初に DACL および第2インターフェイス ACL。そういうわけで明示的な「deny ip any any」を追加する場合、DACL はインターフェイス ACL を考慮に入れません。通常 DACL に明示的な拒否がないし、それからインターフェイス ACL は後適用しますこと。

3750X のバージョン 15.0.2 のための動作は同じですが、明示的 SH IPアクセスリスト interface コマンドはインターフェイス ACL もう示しません (しかしそれまだインターフェイス ACL と否定します存在する DACL で連結されません) を。

802.1X に使用するデフォルトACL

デフォルト ACL には 2 つの型があります:

- auth デフォルト ACL 開いた-開いたモードのために使用されて
- 閉じるアクセスに使用する auth デフォルト ACL -

ポートが不正な状態にあるとき auth デフォルト ACL および auth デフォルト ACL 開いた両方使用されます。デフォルトで、閉じられたアクセスは使用されます。それは 1 つが auth デフォルト ACL によって許可した以外認証がすべてのトラフィック廃棄される前にことを意味します。この方法 DHCP トラフィックは認証の成功の前に許可されます。IP アドレスは割り当てられ、ダウンロードされた DACL は正しく適用することができます。ACL は自動的に作成され、設定で見つけることができないこと。

```
bsns-3750-5#sh run | i Auth-Default
```

```
bsns-3750-5#sh ip access-lists Auth-Default-ACL
Extended IP access list Auth-Default-ACL
 10 permit udp any range bootps 65347 any range bootpc 65348 (22 matches)
 20 permit udp any any range bootps 65347 (12 matches)
 30 deny ip any any
```

それは最初の認証のために最後のセッションが取除かれた後 (認証 および 権限フェーズ間で) 動的に作成され、取除かれます。

Auth デフォルト ACL 割り当て DHCP トラフィックだけ。認証は成功した、新しい DACL はダウンロードされる後、そのセッションに適用されます。auth デフォルト ACL 開いた開くためにモードが現われるおよび変更されるとき同じように使用され、Auth デフォルト ACL としてはたります:

```
bsns-3750-5(config)#int g1/0/2
bsns-3750-5(config-if)#authentication open
```

```
bsns-3750-5#show ip access-lists
Extended IP access list Auth-Default-ACL-OPEN
 10 permit ip any any
```

ACL は両方ともカスタマイズすることができますが設定で決して見られません。

```
bsns-3750-5(config)#ip access-list extended Auth-Default-ACL
bsns-3750-5(config-ext-nacl)#permit udp any any
```

```
bsns-3750-5#sh ip access-lists
Extended IP access list Auth-Default-ACL
 10 permit udp any range bootps 65347 any range bootpc 65348 (22 matches)
 20 permit udp any any range bootps 65347 (16 matches)
 30 deny ip any any
 40 permit udp any any
```

```
bsns-3750-5#sh run | i Auth-Def
bsns-3750-5#
```

モードを開いて下さい

前のセクションは (開いたモードにデフォルトで使用する 1 つが含まれている) のための動作を ACL 記述しました。開いたモードのための動作は次のとおりです:

- それはすべてのトラフィックをセッションが不正な状態にあるとき可能にします (auth デフォルト ACL 開いたデフォルトによって)。
- セッションは認証/許可の間に不正な状態に (暗号化 アプライアンス モデル E (PXE) プー

トシナリオのために) ありますよまたは後そのプロセスは失敗します (「低い影響モード」 と呼ばれるシナリオのためによい)。

- セッションが多重プラットフォームのための Authorized State に移動するとき、ACL は連結され、最初の DACL は、インターフェイス ACL 使用されます。
- 複数の auth またはマルチドメインのためにさまざまな状態に同時にマルチセッションがあるかもしれません (それから別の ACL 型は各セッションに適用します)。

インターフェイス ACL が必須である時

倍数に関しては 6500/4500 のプラットフォームは、インターフェイス ACL DACL を正しく適用して必須です。

4500 sup2 12.2.53SG6 の例は、インターフェイス ACL ここにありません:

```
brisk#show run int g2/3
!
interface GigabitEthernet2/3
  switchport mode access
  switchport voice vlan 10
  authentication host-mode multi-auth
  authentication open
  authentication order mab dot1x
  authentication priority dot1x mab
  authentication port-control auto
  mab
```

それからホストが認証された後、DACL はダウンロードされます。それは適用しないし、許可は失敗します。

```
*Apr 25 04:38:05.239: RADIUS: Received from id 1645/19 10.48.66.74:1645,Access-Accept,
len 209
*Apr 25 04:38:05.239: RADIUS: authenticator 35 8E 59 E4 D5 CF 8F 9A -
EE 1C FC 5A 9F 67 99 B2
*Apr 25 04:38:05.239: RADIUS: User-Name [1] 41
"#ACSACL#-IP-PERMIT_ALL_TRAFFIC-51ef7db1"
*Apr 25 04:38:05.239: RADIUS: State [24] 40
*Apr 25 04:38:05.239: RADIUS: 52 65 61 75 74 68 53 65 73 73 69 6F 6E 3A 30 61
[ReauthSession:0a]
*Apr 25 04:38:05.239: RADIUS: 33 30 34 32 34 61 30 30 30 45 46 35 30 46 35 33
[30424a000EF50F53]
*Apr 25 04:38:05.239: RADIUS: 35 41 36 36 39 33 [ 5A6693]
*Apr 25 04:38:05.239: RADIUS: Class [25] 54
*Apr 25 04:38:05.239: RADIUS: 43 41 43 53 3A 30 61 33 30 34 32 34 61 30 30 30
[CACS:0a30424a000]
*Apr 25 04:38:05.239: RADIUS: 45 46 35 30 46 35 33 35 41 36 36 39 33 3A 69 73
[EF50F535A6693:is]
*Apr 25 04:38:05.239: RADIUS: 65 32 2F 31 38 30 32 36 39 35 33 38 2F 31 32 38
[e2/180269538/128]
*Apr 25 04:38:05.239: RADIUS: 36 35 35 33 [ 6553]
*Apr 25 04:38:05.239: RADIUS: Message-Authenticato[80] 18
*Apr 25 04:38:05.239: RADIUS: AF 47 E2 20 65 2F 59 39 72 9A 61 5C C5 8B ED F5
[ G e/Y9ra\]
*Apr 25 04:38:05.239: RADIUS: Vendor, Cisco [26] 36
*Apr 25 04:38:05.239: RADIUS: Cisco AVpair [1] 30
"ip:inacl#1=permit ip any any"
*Apr 25 04:38:05.239: RADIUS(00000000): Received from id 1645/19
*Apr 25 04:38:05.247: EPM_SESS_ERR:Failed to apply ACL to interface
*Apr 25 04:38:05.247: EPM_API:In function epm_send_message_to_client
*Apr 25 04:38:05.247: EPM_SESS_EVENT:Sending response message to process
```

AUTH POLICY Framework

```
*Apr 25 04:38:05.247: EPM_SESS_EVENT:Returning feature config
*Apr 25 04:38:05.247: EPM_API:In function epm_acl_feature_free
*Apr 25 04:38:05.247: EPM_API:In function epm_policy_aaa_response
*Apr 25 04:38:05.247: EPM_FSM_EVENT:Event epm_ip_wait_event state changed from
policy-apply to ip-wait
*Apr 25 04:38:05.247: EPM_API:In function epm_session_action_ip_wait
*Apr 25 04:38:05.247: EPM_API:In function epm_send_ipwait_message_to_client
*Apr 25 04:38:05.247: EPM_SESS_ERR:NULL feature list for client ctx 1B2694B0
for type DOT1X
*Apr 25 04:38:05.247: %AUTHMGR-5-FAIL: Authorization failed for client
(0007.5032.6941) on Interface Gi2/3
AuditSessionID 0A304345000000060012C050
```

brisk#show authentication sessions

Interface	MAC Address	Method	Domain	Status	Session ID
Gi2/3	0007.5032.6941	mab	VOICE	Authz Failed	0A304345000000060012C050

インターフェイスの後に ACL は追加されません:

```
brisk#show ip access-lists all
Extended IP access list all
 10 permit ip any any (63 matches)
```

```
brisk#sh run int g2/3
!
interface GigabitEthernet2/3
 switchport mode access
 switchport voice vlan 10
 ip access-group all in
 authentication host-mode multi-auth
 authentication open
 authentication order mab dot1x
 authentication priority dot1x mab
 authentication port-control auto
 mab
```

認証 および 権限は成功し、DACL は正しく適用されます:

brisk#show authentication sessions

Interface	MAC Address	Method	Domain	Status	Session ID
Gi2/3	0007.5032.6941	mab	VOICE	Authz Success	0A30434500000008001A2CE4

動作は「開いた」認証に依存していません。DACLを受け入れるために、両方のためのインターフェイス ACL を開きました/閉じましたモードを必要とします。

4500/6500 の DACL

4500/6500 で、DACL は acl_snoop DACLs と適用されます。4500 sup2 12.2.53SG6 (電話 + PC) の例はここに示されています。音声 (10) およびデータ (100) VLAN のための別の ACL があります:

```
brisk#show ip access-lists
Extended IP access list acl_snoop_Gi2/3_10
 10 permit ip host 192.168.2.200 any
 20 deny ip any any
Extended IP access list acl_snoop_Gi2/3_100
 10 permit ip host 192.168.10.12 any
 20 deny ip any any
```

ACL は IPDT に正しいエントリがあるので特定です:

```
brisk#show ip device tracking all
IP Device Tracking = Enabled
IP Device Tracking Probe Count = 3
IP Device Tracking Probe Interval = 30
IP Device Tracking Probe Delay Interval = 0
```

```
-----
IP Address      MAC Address    Vlan  Interface          STATE
-----
192.168.10.12   0007.5032.6941 100   GigabitEthernet2/3  ACTIVE
192.168.2.200   000c.29d7.0617 10    GigabitEthernet2/3  ACTIVE
```

認証されたセッションはアドレスを確認します:

```
brisk#show authentication sessions int g2/3
      Interface: GigabitEthernet2/3
      MAC Address: 000c.29d7.0617
      IP Address: 192.168.2.200
      User-Name: 00-0C-29-D7-06-17
      Status: Authz Success
      Domain: VOICE
      Oper host mode: multi-auth
      Oper control dir: both
      Authorized By: Authentication Server
      Vlan Policy: N/A
      Session timeout: N/A
      Idle timeout: N/A
      Common Session ID: 0A3043450000003003258E0C
      Acct Session ID: 0x00000034
      Handle: 0x54000030
```

Runnable methods list:

```
Method  State
mab     Authc Success
dot1x   Not run
```

```
-----
      Interface: GigabitEthernet2/3
      MAC Address: 0007.5032.6941
      IP Address: 192.168.10.12
      User-Name: 00-07-50-32-69-41
      Status: Authz Success
      Domain: DATA
      Oper host mode: multi-auth
      Oper control dir: both
      Authorized By: Authentication Server
      Vlan Policy: N/A
      Session timeout: N/A
      Idle timeout: N/A
      Common Session ID: 0A3043450000002E031D1DB8
      Acct Session ID: 0x00000032
      Handle: 0x4A00002E
```

Runnable methods list:

```
Method  State
mab     Authc Success
dot1x   Not run
```

この段階では PC および電話は両方 ICMP エコー、インターフェイス ACL 提供にだけ応答します:

```
brisk#show ip access-lists interface g2/3
permit ip host 192.168.10.12 any
```

これは、なぜですか。 DACL が電話のためにだけ押されたので (192.168.10.12)。 PC に関して

は、開いたモードでのインターフェイス ACL は使用されます:

```
interface GigabitEthernet2/3
 ip access-group all in
 authentication open
```

```
brisk#show ip access-lists all
Extended IP access list all
 10 permit ip any any (73 matches)
```

要約すると、acl_snoop は PC および電話両方のために作成されますが、DAACL は電話のためにちょうど戻ります。そういうわけでその ACL はインターフェイスに binded ように見られます。

802.1X のための MAC アドレス ステータス

802.1X 認証が開始するとき、MAC アドレスはまだダイナミックとして見られますが、そのパケットのための操作はドロップするです:

```
bsns-3750-5#show authentication sessions
```

Interface	MAC Address	Method	Domain	Status	Session ID
Gi1/0/1	0007.5032.6941	dot1x	UNKNOWN	Running	C0A8000100000596479F4DCE

```
bsns-3750-5#show mac address-table interface g1/0/1
Mac Address Table
```

```
-----
Vlan    Mac Address      Type             Ports
-----
 100    0007.5032.6941  DYNAMIC         Drop
```

Total Mac Addresses for this criterion: 1

認証の成功が MAC アドレスなった後ステイックおよびポート番号は提供されます:

```
bsns-3750-5#show authentication sessions
```

Interface	MAC Address	Method	Domain	Status	Session ID
Gi1/0/1	0007.5032.6941	mab	VOICE	Authz Success	C0A8000100000596479F4DCE

```
bsns-3750-5#show mac address-table interface g1/0/1
Mac Address Table
```

```
-----
Vlan    Mac Address      Type             Ports
-----
 100    0007.5032.6941  STATIC          Gi1/0/1
```

それは両方のドメイン (音声/データ) のすべての mab/dot1x セッションにあてはまります。

トラブルシューティング

特定のソフトウェア バージョンおよびプラットフォームのための 802.1X コンフィギュレーション ガイドを読むことを忘れないようにして下さい。

TAC ケースをオープンする場合、これらのコマンドからの出力を提供して下さい:

- show tech
- show authentication セッション インターフェイス <xx> 詳細
- show mac アドレス テーブル インターフェイス <xx>

SPAN ポート パケットキャプチャおよびこれらのデバッグを収集することもまたよいです:

- debug radius 詳細表示
- デバッグ epm すべて
- debug authentication すべて
- デバッグ dot1x すべて
- debug authentication 機能 <yy> すべて
- debug aaa authentication
- debug aaa authorization

関連情報

- [802.1X 認証サービス コンフィギュレーション ガイド、Cisco IOS XE リリース 3SE \(Catalyst 3850 スイッチ \)](#)
- [Catalyst 3750-X および Catalyst 3560-X スイッチ ソフトウェア コンフィギュレーション ガイド、Cisco IOS Release 15.2\(1\)E](#)
- [Catalyst 3750-X および 3560-X ソフトウェア コンフィギュレーション ガイド、リリース 15.0\(1\)SE](#)
- [Catalyst 3560 ソフトウェア コンフィギュレーション ガイド、リリース 12.2\(52\)SE](#)
- [テクニカルサポートとドキュメント - Cisco Systems](#)