

Dépannage d'IBNS 2.0

Contenu

[Introduction](#)

[Conditions préalables](#)

[Conditions requises](#)

[Components Used](#)

[Informations générales](#)

[Configuration](#)

[Dépannage](#)

[debug mab all](#)

[debug dot1x all](#)

[debug radius](#)

[debug aaa authentication/authentication](#)

[Informations connexes](#)

Introduction

Ce document décrit la procédure de dépannage des authentifications sur les commutateurs qui utilisent IBNS (Identity-Based Networking Services) 2.0

Conditions préalables

Conditions requises

Cisco vous recommande de prendre connaissance des rubriques suivantes :

- Identity Service Engine (ISE)
- Concepts IEEE 802.1X (dot1X)
- Contournement d'authentification MAC (MAB)

Components Used

Les informations de ce document sont basées sur ces versions de logiciel et de matériel, mais pas uniquement sur :

- Commutateur Cisco - C3750X-48PF-S avec IOS 15.2.1E3(ED)
- Identity Service Engine 2.1

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. If your network is live, make sure that you understand the potential impact of any command.

Informations générales

IBNS 2.0 est un nouveau moteur de stratégie qui remplace le gestionnaire d'authentification traditionnel. Il est équipé d'un ensemble de fonctionnalités améliorées qui offrent une configuration flexible avec le langage de politique de classification commune (C3PL) de Cisco. Maintenant appelé Access Session Manager, IBNS 2.0 offre aux administrateurs des options pour configurer les politiques et les actions en fonction de conditions spécifiques et d'événements de point de terminaison. Au lieu de conditions normales, C3PL est utilisé pour définir les conditions d'authentification, les paramètres et les actions. Pour plus d'informations sur IBNS 2.0, suivez le lien fourni dans la section Informations connexes.

Il existe différents types de cartes de politiques qui sont utilisées à diverses fins. Ce paragraphe se concentre sur le type d'abonné. Il y a trois sections dans une carte des politiques à noter.

- Section Événement
- Section Classe
- Section Action

Ils suivent la hiérarchie **Event > Class > Action**. Lorsqu'une carte de stratégie est appliquée à une interface, tous les événements définis dans la carte de stratégie sont évalués. En fonction de l'événement en cours, l'action appropriée définie dans la carte de stratégie est appliquée au niveau de l'interface.

Une fois l'événement associé, il est possible d'évaluer les classes en fonction de l'événement/méthode/résultat de l'authentification/autorisation. Les résultats de ces classes peuvent être **TOUJOURS EXÉCUTÉS** ou appelés dans des cartes de classe supplémentaires.

Dans la section Action, les actions importantes qui peuvent être incluses sont les suivantes :

- Spécifier une méthode d'authentification avec une priorité

```
event session-started match-all
  10 class do-until-failure 10 authenticate using priority
```

- Spécifier une liste de méthodes d'authentification pour une méthode d'authentification particulière

```
event session-started match-all
  10 class do-until-failure 10 authenticate using aaa authc-list
```

- Spécifier une liste de méthodes d'autorisation pour une méthode d'authentification

```
event session-started match-all
  10 class do-until-failure 10 authenticate using aaa authz-list
```

- Spécifier le nombre de tentatives

```
event session-started match-all
  10 class do-until-failure 10 authenticate using retries
```

- Remplacer les données d'authentification/autorisation existantes par de nouvelles données d'authentification/autorisation

```
event timer-expiry match-all
```

```
10 class do-until-failure 10 authenticate using replace aaa
```

- Autorisation de force

```
event session-started match-all  
10 class do-until-failure 10 authorize
```

- Forcer la non-autorisation

```
event timer-expiry match-all  
10 class do-until-failure 10 unauthorize
```

- Activer un modèle de service

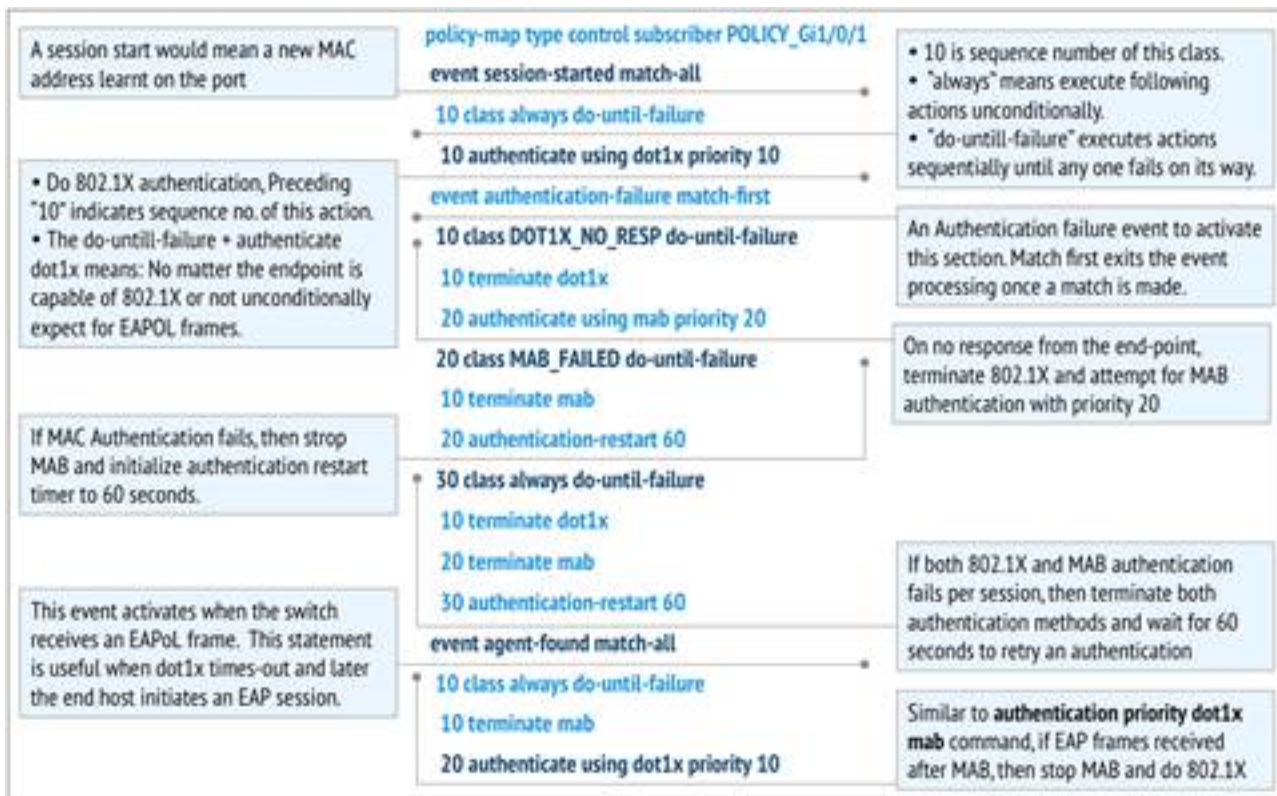
```
event timer-expiry match-all  
10 class do-until-failure 10 activate service-template
```

Dans les commutateurs IOS traditionnels, il n'y avait aucune option pour appliquer une liste de méthodes spécifique à une session authentifiée. IBNS 2.0 fournit cette fonctionnalité à l'aide de modèles de service. Le modèle de service est configuré localement sur le commutateur et appliqué après l'autorisation de session réussie. Il est également possible de pousser le modèle de service requis à partir d'un serveur AAA.

L'attribut radius utilisé pour faire la même chose est *abonné* : *nom-service = <nom du modèle de service>*. Dans Identity Service Engine (ISE), vous pouvez nommer le profil d'autorisation exactement comme le modèle de service local configuré sur le commutateur et cocher la case *Modèle de service*. Ce profil d'autorisation ainsi que tout autre profil d'autorisation peuvent être affichés comme résultat d'autorisation.

Dans le rapport de résultats d'autorisation, il y a une paire AV-Cisco nommée *abonné* : *nom-service = <nom du modèle de service>*. Cela indique que le commutateur a été averti d'appliquer ce modèle de service pour cette session.

Voici une image qui montre la signification exacte de chaque entité d'un exemple de carte de politique.



Configuration

Configuration AAA

```

aaa new-model
aaa authentication dot1x default group radius
aaa authorization exec default local
aaa authorization network default group radius
aaa accounting identity default start-stop group radius
aaa session-id common

```

```
dot1x system-auth-control
```

Configuration du serveur RADIUS

```

radius server ise
  address ipv4 X.X.X.X auth-port 1812 acct-port 1813
  automate-tester username probe-user
  key XXXXXXXXXXXX

```

Configuration de la carte de stratégie

```

policy-map type control subscriber Inter_Gi_3/0/48
  event session-started match-all //On session-start event 10 class always do-until-failure //Both mab and dot1x start at the same time 10 authenticate using dot1x priority 10 20
  authenticate using mab priority 20 event authentication-failure match-first //On authentication event failure 10 class DOT1X_NO_RESP do-until-failure //If dot1x fails 10 terminate dot1x 20
  authenticate using mab priority 20 20 class MAB_FAILED do-until-failure //If mab fails 10
  terminate mab 20 authentication-restart 60 30 class always do-until-failure //If both mab and dot1x fail 10 terminate dot1x 20 terminate mab 30 authentication-restart 60 event agent-found
  match-all //On dot1x agent found event 10 class always do-until-failure 10 terminate mab 20
  authenticate using dot1x priority 10

```

Configuration des cartes de classe

```
class-map type control subscriber match-all DOT1X_NO_RESP //If dot1x and no response from client
match method dot1x match result-type method dot1x agent-not-found
class-map type control subscriber match-all MAB_FAILED //On mab failure match method mab match
result-type method mab authoritative
```

Configuration de l'interface

```
interface GigabitEthernet3/0/48
description ** Access Port **
switchport access vlan 100
switchport mode access
switchport voice vlan 10
ip access-group IPV4-PRE-AUTH-ACL in
access-session port-control auto
mab
dot1x pae authenticator
spanning-tree portfast
service-policy type control subscriber Inter_Gi_3/0/48
```

Dépannage

Le meilleur moyen de dépanner est de comparer les journaux de travail et les journaux de non-travail. De cette manière, l'étape exacte à laquelle le processus a mal tourné est connue. Il y a quelques débogages qui doivent être activés pour résoudre les problèmes mab/dot1x. Voici les commandes permettant d'activer ces débogages.

- debug aaa authentication
- debug aaa authorization
- debug mab all
- debug dot1x all
- debug radius

Voici les journaux de travail avec dot1x et mab activés en même temps.

debug mab all

```
mab-ev: [28d2.4496.5376, Gi3/0/48] Received MAB context create from AuthMgr // New mac-address
detected mab-ev: MAB authorizing 28d2.4496.5376 //mab authorization event should start mab-ev:
Created MAB client context 0xB0000001 mab : initial state mab_initialize has enter //Initialize
mab mab-ev: [28d2.4496.5376, Gi3/0/48] Sending create new context event to EAP from MAB for
0xB0000001 (28d2.4496.5376) mab-ev: [28d2.4496.5376, Gi3/0/48] MAB authentication started for
0x0782A870 (28d2.4496.5376) //mab authentication initialized %AUTHMGR-5-START: Starting 'mab'
for client (28d2.4496.5376) on Interface Gi3/0/48 AuditSessionID 0A6A258E0000003300C586C2 mab-
ev: [28d2.4496.5376, Gi3/0/48] Invalid EVT 9 from EAP mab-sm: [28d2.4496.5376, Gi3/0/48]
Received event 'MAB_CONTINUE' on handle 0xB0000001 mab : during state mab_initialize, got event
1(mabContinue) @@@ mab : mab_initialize -> mab_authorizing //mab authorizing event started mab-
ev: [28d2.4496.5376] formatted mac = 28d244965376 //mac-address formatted as required mab-ev:
[28d2.4496.5376] created mab pseudo dot1x profile dot1x_mac_auth_28d2.4496.5376 //peuso dot1x
profile formed (username=macaddress) mab-ev: [28d2.4496.5376, Gi3/0/48] Starting MAC-AUTH-BYPASS
for 0xB0000001 (28d2.4496.5376) //starting mab authentication mab-ev: [28d2.4496.5376, Gi3/0/48]
Invalid EVT 9 from EAP mab-ev: [28d2.4496.5376, Gi3/0/48] MAB received an Access-Accept for
0xB0000001 (28d2.4496.5376) //received mab success from the server %MAB-5-SUCCESS:
Authentication successful for client (28d2.4496.5376) on Interface Gi3/0/48 AuditSessionID
```

```
0A6A258E0000003300C586C2 mab-sm: [28d2.4496.5376, Gi3/0/48] Received event 'MAB_RESULT' on
handle 0xB0000001 // mab authorization result received mab : during state mab_authorizing, got
event 5(mabResult) @@@ mab : mab_authorizing -> mab_terminate //mab authorization process
terminate mab-ev: [28d2.4496.5376, Gi3/0/48] Deleted credentials profile for 0xB0000001
(dot1x_mac_auth_28d2.4496.5376) //deleted pseudo dot1x profile %AUTHMGR-5-SUCCESS: Authorization
succeeded for client (28d2.4496.5376) on Interface Gi3/0/48 AuditSessionID
0A6A258E0000003300C586C2 // posting mab authorization succeeded
```

debug dot1x all

Puisque dot1x a beaucoup d'échanges de messages à cause des négociations de protocole, des échanges de certificats et ainsi de suite, tous les journaux de débogage n'ont pas été mentionnés ici. Le flux d'événements dans l'ordre dans lequel ils sont censés se produire et leurs journaux de débogage correspondants ont été documentés ici.

```
dot1x-packet:EAPOL pak rx - Ver: 0x1 type: 0x1 // Initial EAPoL packet received by switch
dot1x-packet: length: 0x0000 dot1x-ev:[28d2.4496.5376, Gi3/0/48] New client detected, sending
session start event for 28d2.4496.5376 // dot1x client detected dot1x-ev:[28d2.4496.5376,
Gi3/0/48] Dot1x authentication started for 0x26000007 (28d2.4496.5376) //dot1x started %AUTHMGR-
5-START: Starting 'dot1x' for client (28d2.4496.5376) on Interface Gi3/0/48 AuditSessionID
0A6A258E0000003500C9CFC3 dot1x-sm:[28d2.4496.5376, Gi3/0/48] Posting !EAP_RESTART on Client
0x26000007 //requesting client to restart the EAP Proces dot1x-sm:[28d2.4496.5376, Gi3/0/48]
Posting RX_REQ on Client 0x26000007 //waiting fot the EAPoL packet fromt he client dot1x-
sm:[28d2.4496.5376, Gi3/0/48] Posting AUTH_START for 0x26000007 // Starting authentication
process dot1x-ev:[28d2.4496.5376, Gi3/0/48] Sending out EAPOL packet // Identity Request dot1x-
packet:EAPOL pak Tx - Ver: 0x3 type: 0x0 dot1x-packet: length: 0x0005 dot1x-packet:EAP code: 0x1
id: 0x1 length: 0x0005 dot1x-packet: type: 0x1 dot1x-packet:[28d2.4496.5376, Gi3/0/48] EAPOL
packet sent to client 0x26000007 dot1x-ev:[Gi3/0/48] Received pkt saddr =28d2.4496.5376 , daddr
= 0180.c200.0003, pae-ether-type = 888e.0100.000a dot1x-packet:EAPOL pak rx - Ver: 0x1 type: 0x0
// Identity Response dot1x-packet: length: 0x000A dot1x-sm:[28d2.4496.5376, Gi3/0/48] Posting
EAPOL_EAP for 0x26000007 //EAPoL packet(EAP Response) received, preparing request to server
dot1x-sm:[28d2.4496.5376, Gi3/0/48] Posting EAP_REQ for 0x26000007 //Server response received,
EAP Request is being prepared dot1x-ev:[28d2.4496.5376, Gi3/0/48] Sending out EAPOL packet
dot1x-packet:EAPOL pak Tx - Ver: 0x3 type: 0x0 dot1x-packet: length: 0x0006 dot1x-packet:EAP
code: 0x1 id: 0xE5 length: 0x0006 dot1x-packet: type: 0xD dot1x-packet:[28d2.4496.5376,
Gi3/0/48] EAPOL packet sent to client 0x26000007 //EAP request sent out dot1x-ev:[Gi3/0/48]
Received pkt saddr =28d2.4496.5376 , daddr = 0180.c200.0003, pae-ether-type = 888e.0100.0006
//EAP response received dot1x-packet:EAPOL pak rx - Ver: 0x1 type: 0x0 dot1x-packet: length:
0x0006 || || || || Here a lot of EAPOL-EAP and EAP_REQ events occur as a lot of information is
exchanged between the switch and the client
|| If the events after this do not follow, then the timers and the information sent till now
need to be checked || || || dot1x-packet:[28d2.4496.5376, Gi3/0/48] Received an EAP Success
//EAP Success recieved from Server dot1x-sm:[28d2.4496.5376, Gi3/0/48] Posting EAP_SUCCESS for
0x26000007 //Posting EAP Success event dot1x-sm:[28d2.4496.5376, Gi3/0/48] Posting AUTH_SUCCESS
on Client 0x26000007 //Posting Authentication success %DOT1X-5-SUCCESS: Authentication
successful for client (28d2.4496.5376) on Interface Gi3/0/48 AuditSessionID
0A6A258E0000003500C9CFC3
dot1x-packet:[28d2.4496.5376, Gi3/0/48] EAP Key data detected adding to attribute list
//Additional key data detected sent by server
%AUTHMGR-5-SUCCESS: Authorization succeeded for client (28d2.4496.5376) on Interface Gi3/0/48
AuditSessionID 0A6A258E0000003500C9CFC3 dot1x-ev:[28d2.4496.5376, Gi3/0/48] Received Authz
Success for the client 0x26000007 (28d2.4496.5376) //Authorization Success dot1x-
ev:[28d2.4496.5376, Gi3/0/48] Sending out EAPOL packet //Sending EAP Success to the client
dot1x-packet:EAPOL pak Tx - Ver: 0x3 type: 0x0 dot1x-packet: length: 0x0004 dot1x-packet:EAP
code: 0x3 id: 0xED length: 0x0004 dot1x-packet:[28d2.4496.5376, Gi3/0/48] EAPOL packet sent to
client 0x26000007
```

debug radius

Comme il y a beaucoup de messages EAP, les paquets RADIUS envoyés au serveur et reçus seront également plus nombreux. Toutes les authentifications dot1x ne se terminent pas par sur

Access-Request. Par conséquent, les journaux présentés ici sont ceux qui sont importants et au fur et à mesure du flux.

```
//mab and dot1x start at the same time as per the configuration
%AUTHMGR-5-START: Starting 'dot1x' for client (28d2.4496.5376) on Interface Gi3/0/48
AuditSessionID 0A6A258E0000003600CCC037 %AUTHMGR-5-START: Starting 'mab' for client
(28d2.4496.5376) on Interface Gi3/0/48 AuditSessionID 0A6A258E0000003600CCC037
RADIUS/ENCODE(00000000):Orig. component type = Invalid RADIUS(00000000): Config NAS IP: 0.0.0.0
//Since dot1x client didn't respond yet, mab authentication is done
RADIUS(00000000): sending RADIUS/ENCODE: Best Local IP-Address 10.106.37.142 for Radius-Server
10.106.73.143 RADIUS(00000000): Send Access-Request to 10.106.73.143:1812 id 1645/56, len 267
RADIUS: authenticator F0 E4 E3 28 7E EA E6 83 - 43 55 7F DC 96 19 EB 42 RADIUS: User-Name [1] 14
"28d244965376" RADIUS: User-Password [2] 18 * RADIUS: Service-Type [6] 6 Call Check [10] RADIUS:
Vendor, Cisco [26] 31 RADIUS: Cisco AVpair [1] 25 "service-type=Call Check" RADIUS: Framed-MTU
[12] 6 1500 RADIUS: Called-Station-Id [] 19 "CC-EF-48-AD-6B-" RADIUS: Calling-Station-Id [31] 19
"28-D2-44-96-53-76" RADIUS: Message-Authenticato[80] 18 RADIUS: AD DC 22 D7 83 8C 02 C5 1E 11 B2
94 80 85 2F 3D [ "/=] RADIUS: EAP-Key-Name [102] 2 * RADIUS: Vendor, Cisco [26] 49 RADIUS: Cisco
AVpair [1] 43 "audit-session-id=0A6A258E0000003600CCC037" RADIUS: Vendor, Cisco [26] 18 RADIUS:
Cisco AVpair [1] 12 "method=mab" RADIUS: Framed-IP-Address [8] 6 1.1.1.2 RADIUS: NAS-IP-Address
[4] 6 10.106.37.142 RADIUS: NAS-Port [5] 6 60000 RADIUS: NAS-Port-Id [87] 23
"GigabitEthernet3/0/48" RADIUS: NAS-Port-Type [61] 6 Ethernet [15] RADIUS(00000000): Sending a
IPv4 Radius Packet RADIUS(00000000): Started 5 sec timeout RADIUS: Received from id 1645/56
10.106.73.143:1812, Access-Accept, len 176 RADIUS: authenticator 7B D6 DA E1 70 49 6E 6D - 3D AC
5C 1D C0 AC CF D0 RADIUS: User-Name [1] 19 "28-D2-44-96-53-76" RADIUS: State [24] 40 RADIUS: 52
65 61 75 74 68 53 65 73 73 69 6F 6E 3A 41 [ReauthSession:0A] RADIUS: 36 41 32 35 38 45 33 36
[6A258E0000003600] RADIUS: 43 43 43 33 37 [ CCC037] RADIUS: Class [25] 51 RADIUS: 43 41 43 53 3A
41 36 41 32 35 38 45 [CACS:0A6A258E000] RADIUS: 33 36 43 43 43 33 37 3A 69 73 [0003600CCC037:is]
RADIUS: 65 31 34 2F 32 35 35 38 35 37 38 34 2F 36 34 [e14/255857804/64] RADIUS: 36 [ 6] RADIUS:
Message-Authenticato[80] 18 RADIUS: D3 F3 6E 9A 25 09 01 8C D6 B1 20 D6 84 D3 18 3D [ n? =]
RADIUS: Vendor, Cisco [26] 28 RADIUS: Cisco AVpair [1] 22 "profile-name=Unknown" //mab succeeds
%MAB-5-SUCCESS: Authentication successful for client (28d2.4496.5376) on Interface Gi3/0/48
AuditSessionID 0A6A258E0000003600CCC037 %AUTHMGR-5-SUCCESS: Authorization succeeded for client
(28d2.4496.5376) on Interface Gi3/0/48 AuditSessionID 0A6A258E0000003600CCC037 //A dot1x client
is detected and mab is stopped as per the configuration and dot1x authentication starts
%AUTHMGR-7-STOPPING: Stopping 'mab' for client 28d2.4496.5376 on Interface Gi3/0/48
AuditSessionID 0A6A258E0000003600CCC037 RADIUS/ENCODE(00000000):Orig. component type = Invalid
RADIUS(00000000): Config NAS IP: 0.0.0.0 RADIUS(00000000): sending RADIUS/ENCODE: Best Local IP-
Address 10.106.37.142 for Radius-Server 10.106.73.143 RADIUS(00000000): Send Access-Request to
10.106.73.143:1812 id 1645/57, len 252 RADIUS: authenticator 1B E9 37 F4 AC C7 73 BE - F4 95 CB
5F FC 2D 3D E1 RADIUS: User-Name [1] 7 "cisco" RADIUS: Service-Type [6] 6 Framed [2] RADIUS:
Vendor, Cisco [26] 27 RADIUS: Cisco AVpair [1] 21 "service-type=Framed" RADIUS: Framed-MTU [12]
6 1500 RADIUS: Called-Station-Id [] 19 "CC-EF-48-AD-6B-" RADIUS: Calling-Station-Id [31] 19 "28-
D2-44-96-53-76" RADIUS: EAP-Message [79] 12 RADIUS: 02 01 00 0A 01 63 69 73 63 6F [ cisco]
RADIUS: Message-Authenticato[80] 18 RADIUS: 7B 42 C2 C2 69 CB 73 49 1A 40 81 28 71 CF CC 86 [
{BisI@q] RADIUS: EAP-Key-Name [102] 2 * RADIUS: Vendor, Cisco [26] 49 RADIUS: Cisco AVpair [1]
43 "audit-session-id=0A6A258E0000003600CCC037" RADIUS: Vendor, Cisco [26] 20 RADIUS: Cisco
AVpair [1] 14 "method=dot1x" RADIUS: Framed-IP-Address [8] 6 1.1.1.2 RADIUS: NAS-IP-Address [4]
6 10.106.37.142 RADIUS: NAS-Port [5] 6 60000 RADIUS: NAS-Port-Id [87] 23 "GigabitEthernet3/0/48"
RADIUS: NAS-Port-Type [61] 6 Ethernet [15] RADIUS(00000000): Sending a IPv4 Radius Packet //More
information is being requested by the AAA Server RADIUS: Received from id 1645/57
10.106.73.143:1812, Access-Challenge, len 120 RADIUS: authenticator A7 2A 6E 8C 75 9C 28 6F - 32
85 B9 87 5B D2 E4 FB RADIUS: State [24] 74 RADIUS: 33 37 43 50 4D 53 65 73 73 69 6F 6E 49 44 3D
[37CPMSessionID=0] RADIUS: 41 36 41 32 35 38 45 33 36 [A6A258E000000360] RADIUS: 43 43 43 33 37
3B 32 39 53 65 73 73 69 6F [0CCC037;29Sessio] RADIUS: 6E 49 44 3D 69 73 65 31 34 2F 32 35 35 38
35 37 [nID=ise14/255857] RADIUS: 38 34 2F 36 34 38 3B [ 804/648;] RADIUS: EAP-Message [79] 8
RADIUS: 01 0A 00 06 0D 20 [ ] RADIUS: Message-Authenticato[80] 18 RADIUS: E2 7C 2B 0E CA AB E3
21 B8 CD 04 8A 7F 23 7A D2 [ |+!#z] || || || || || As mentioned before, the excess logs of Access-
Requestes and Access-Challenges come here || || || //Authentication and Authorization succeeds
for dot1x
RADIUS: Received from id 1645/66 10.106.73.143:1812, Access-Accept, len 325 RADIUS:
authenticator F0 CF EE 59 3A 26 25 8F - F7 0E E4 03 E1 11 7E 86 RADIUS: User-Name [1] 7 "cisco"
RADIUS: State [24] 40 RADIUS: 52 65 61 75 74 68 53 65 73 73 69 6F 6E 3A 41 [ReauthSession:0A]
```

```
RADIUS: 36 41 32 35 38 45 33 36 [6A258E0000003600] RADIUS: 43 43 43 33 37 [ CCC037] RADIUS:
Class [25] 51 RADIUS: 43 41 43 53 3A 41 36 41 32 35 38 45 [CACs:0A6A258E000] RADIUS: 33 36 43 43
43 33 37 3A 69 73 [0003600CCC037:is] RADIUS: 65 31 34 2F 32 35 35 38 35 37 38 34 2F 36 34
[e14/255857804/64] RADIUS: 38 [ 8] RADIUS: EAP-Message [79] 6 RADIUS: 03 12 00 04 RADIUS:
Message-Authenticato[80] 18 RADIUS: 3F 7A DA 59 F7 8A DE 1D 33 4B 07 88 62 F3 3B 71 [ ?zY3Kb;q]
RADIUS: EAP-Key-Name [102] 67 * RADIUS: Vendor, Microsoft [26] 58 RADIUS: MS-MPPE-Send-Key [16]
52 * RADIUS: Vendor, Microsoft [26] 58 RADIUS: MS-MPPE-Recv-Key [17] 52 * RADIUS(00000000):
Received from id 1645/66 RADIUS/DECODE: EAP-Message fragments, 4, total 4 bytes //Dot1x succeeds
%DOT1X-5-SUCCESS: Authentication successful for client (28d2.4496.5376) on Interface Gi3/0/48
AuditSessionID 0A6A258E0000003600CCC03
```

debug aaa authentication/authentication

l'authentification aaa debug et l'autorisation aaa debug affichent des informations utiles lors de diverses méthodes d'authentification/autorisation. Dans ce cas, il ne s'agit que d'une seule ligne spécifiant la liste de méthodes utilisée.

```
AAA/AUTHEN/8021X (00000000): Pick method list 'default'
```

Indique si l'une des méthodes d'authentification n'est pas disponible/n'est pas activée.

La procédure de dépannage des CWA/Posture/DACL, etc., est identique à celle des commutateurs IOS traditionnels. La vérification de la configuration est la première étape du dépannage. Assurez-vous que la configuration répond aux exigences. Si la configuration de la carte de stratégie, la carte de classe est à la hauteur de la marque, alors les problèmes de débogage s'il y en a, peuvent être très faciles. Pour plus d'informations sur la configuration à l'aide d'IBNS 2.0, reportez-vous à la section Informations connexes.

Informations connexes

- [Guide de déploiement IBNS 2.0](#)