Configure IBNS 2.0 for Single-Host and Multi-Domain Scenarios

Contents

Introduction

This document describes how to configure Identity Based Networking Services 2.0 (IBNS) for single-host and multi-domain scenarios.

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- Extensible Authentication Protocol over Local Area Network (EAPoL)
- Radius protocol
- Cisco Identity Services Engine version 2.0

Components Used

The information in this document is based on these software and hardware versions:

- Cisco Identity Service Engine version 2.0 patch 2
- Endpoint with Windows 7 OS
- Cisco switch 3750X with IOS 15.2(4)E1
- Cisco switch 3850 with 03.02.03.SE
- Cisco IP Phone 9971

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, ensure that you understand the potential impact of any command.

Configure

Configuration Theory

In order to enable IBNS 2.0, you need to execute the command in privilege mode on your Cisco switch:

```
#authentication display new-style
```

Configure switchport for IBNS 2.0 with commands as shown:

```
 access-session host-mode {single-host | multi-domain | multi-auth}
 access-session port-control auto
 dot1x pae authenticator
 {mab}
 service-policy type control subscriber TEST
```

These commands enable dot1x authentication, and optionally, MAC Authentication Bypass (MAB) on the interface. When you use the new syntax, you use commands which start with access-session. The purpose of those commands is the same as for commands that use old syntax (starting with authentication keyword). Apply service-policy to specify policy-map that can be used for the interface.

The policy-map mentioned defines behavior of the switch (authenticator) during authentication. For example, you can specify what can happen in case of authentication failure. For each event you can configure multiple actions based on the type of the event matched in class-map configured under it. As an example, take a look at the list as shown (policy-map TEST4). If dot1x endpoint, which is connected to the interface where this policy is applied fails, then action defined in DOT1X_FAILED is executed. If you would like to specify the same behavior for classes like MAB_FAILED and DOT1X_FAILED, then you can use default class - class-map always.

```
policy-map type control subscriber TEST4
(...)
 event authentication-failure match-first
  10 class DOT1X_FAILED do-until-failure
   10 terminate dot1x
(...)
  40 class always do-until-failure
   10 terminate mab
   20 terminate dot1x
   30 authentication-restart 60
(...)
```

Policy-map used for IBNS 2.0 always must have type control subscriber.

You can view the list of available events in this way:

```
Switch(config-event-control-policymap)#event ?
  aaa-available                aaa-available event
  absolute-timeout             absolute timeout event
  agent-found                  agent found event
  authentication-failure       authentication failure event
  authentication-success       authentication success event
  authorization-failure        authorization failure event
  inactivity-timeout           inactivity timeout event
  session-started              session started event
  tag-added                    tag to apply event
  tag-removed                  tag to remove event
  template-activated           template activated event
  template-activation-failed   template activation failed event
  template-deactivated         template deactivated event
  template-deactivation-failed template deactivation failed event
  timer-expiry                 timer-expiry event
  violation                    session violation event
```

In event configuration, you have the possibility to define how classes can be evaluated:

```
Switch(config-event-control-policymap)#event authentication-failure ?
  match-all    Evaluate all the classes
  match-first  Evaluate the first class
```

You can define similar options for class-maps, although here you specify how actions can be executed in case your class is matched:

```
Switch(config-class-control-policymap)#10 class always ?
  do-all            Execute all the actions
  do-until-failure  Execute actions until one of them fails
  do-until-success  Execute actions until one of them is successful
```

Last part (optional) of configuration in new style of dot1x is class-map. It also can type control subscriber, and it is used to match specific behavior or traffic. Configure requirements for class-map condition evaluation. You can specify that all conditions have to be matched, or any condition has to be matched, or none of the conditions match.

```
Switch(config)#class-map type control subscriber ?
  match-all   TRUE if everything matches in the class-map
  match-any   TRUE if anything matches in the class-map
  match-none  TRUE if nothing matches in the class-map
```

This is an example of class-map used for matching dot1x authentication failure:

```
class-map type control subscriber match-all DOT1X_FAILED
 match method dot1x
```

```
 match result-type method dot1x authoritative
```

For some scenarios, mostly when service-template is in use, you need to add configuration for Change of Authorization (CoA):

```
aaa server radius dynamic-author
 client 10.48.17.232 server-key cisco
```

Scenario for Single-Host

Network Diagram

Configurations

Basic 802.1X configuration required for single-host scenario tested on Catalyst 3750X with IOS 15.2(4)E1. Scenario tested with Windows Native Supplicant and Cisco AnyConnect.

```
aaa new-model
!
aaa group server radius tests
 server name RAD-1
!
aaa authentication dot1x default group tests
aaa authorization network default group tests
!
dot1x system-auth-control
!
policy-map type control subscriber TEST
 event session-started match-all
  10 class always do-until-failure
   10 authenticate using dot1x priority 10
!
interface GigabitEthernet1/0/21
 switchport access vlan 613
 switchport mode access
 access-session host-mode single-host
 access-session port-control auto
 dot1x pae authenticator
 service-policy type control subscriber TEST
!
radius server RAD-1
 address ipv4 10.48.17.232 auth-port 1812 acct-port 1813
 key cisco
```

Scenario for Multi-Domain

Network Diagram

Configurations

Multi-domain scenario was tested on Catalyst 3850 with IOS 03.02.03.SE due to PoE (Power over Ethernet) requirements for IP Phone (Cisoc IP Phone 9971).

```
aaa new-model
!
aaa group server radius tests
 server name RAD-1
!
aaa authentication dot1x default group tests
aaa authorization network default group tests
!
aaa server radius dynamic-author
 client 10.48.17.232 server-key cisco
!
dot1x system-auth-control
!
class-map type control subscriber match-all DOT1X
 match method dot1x
!
class-map type control subscriber match-all DOT1X_FAILED
 match method dot1x
 match result-type method dot1x authoritative
!
class-map type control subscriber match-all DOT1X_NO_RESP
 match method dot1x
 match result-type method dot1x agent-not-found
!
class-map type control subscriber match-all MAB
 match method mab
!
class-map type control subscriber match-all MAB_FAILED
 match method mab
 match result-type method mab authoritative
!
policy-map type control subscriber TEST4
 event session-started match-all
  10 class always do-until-failure
   10 authenticate using dot1x priority 10
   20 authenticate using mab priority 20
 event authentication-failure match-first
  10 class DOT1X_FAILED do-until-failure
   10 terminate dot1x
  20 class MAB_FAILED do-until-failure
   10 terminate mab
   20 authenticate using dot1x priority 10
  30 class DOT1X_NO_RESP do-until-failure
```

```
   10 terminate dot1x
   20 authentication-restart 60
  40 class always do-until-failure
   10 terminate mab
   20 terminate dot1x
   30 authentication-restart 60
 event agent-found match-all
  10 class always do-until-failure
   10 terminate mab
   20 authenticate using dot1x priority 10
 event authentication-success match-all
  10 class always do-until-failure
   10 activate service-template DEFAULT_LINKSEC_POLICY_SHOULD_SECURE
!
interface GigabitEthernet1/0/1
 switchport access vlan 613
 switchport mode access
 switchport voice vlan 612
 access-session host-mode multi-domain
 access-session port-control auto
 mab
 dot1x pae authenticator
 spanning-tree portfast
 service-policy type control subscriber TEST4
!
radius-server attribute 6 on-for-login-auth
radius-server attribute 8 include-in-access-req
radius-server attribute 25 access-request include
radius-server vsa send cisco-nas-port
!
radius server RAD-1
 address ipv4 10.48.17.232 auth-port 1812 acct-port 1813
 key cisco
```

Verify

Use this section in order to confirm that your configuration works properly.

For verification purposes, use this command to list sessions from all switchports:

```
show access-session
```

You can also view detailed information about sessions from a single switchport:

```
show access-session interface [Gi 1/0/1] {detail}
```

Troubleshoot

This section provides information you can use in order to troubleshoot your configuration.

In order to troubleshoot 802.1X related issues, you can enable debugs the same way as for old style 802.1X syntax:

```
debug mab all
debug dot1x all
debug pre all*
```

* optionally for debug pre you can use only event and/or rule to limit output to IBNS 2.0 relevant information.