CCNP Security SISAS Notes

Dot1x:

Authentication Modes:

Single-host	One host hanging off port
Multi-host	Multiple devices off port, first device's MAC enables port for all
Multi-domain	Data + voice (each requiring authentication)
Multi-auth	Each individual device must authenticate

Set up AAA:
aaa new-model
aaa authentication dot1x default group radius
aaa authorization network default group radius
aaa accounting dot1x default start-stop group radius

Test AAA:
test aaa group NAME username password new-code

Include supplicant IP address in accounting logs:
radius-server attribute 8 include-in-access-req

Globally enable dot1q:
dot1x system-auth-control

Tip: Use "**switchport host**" as a macro to enable switchport mode access and portfast

Set authentication mode (as listed above):
authentication host-mode multi-auth

Start out easy… set authentication to OPEN, such that even if the supplicant fails authentication the port will still pass traffic:
authentication open << KEEP YOUR JOB ☺

Enable recurring authentication:
authentication periodic
authentication timer reauthenticate server (server decides how often)

Set Port Access Entity (PAE) to act as authentication … switch asks supplicant to identify itself:
dot1x pae authenticator

If authentication fails, wait *x* seconds before trying again:
dot1x timeout tx-period *10*

Tell Dot1x to control the port... Dot1x will determine whether the port is or is not authorized based on the success/failure of the supplicant's authentication:
dot1x port-control auto

A helpful debug to check the authentication status:
debug radius authentication

Show commands:
show dot1x all
show authentication int *gi x/x*
show authentication sessions
show authentication sessions int *gi x/x*

MAC Authentication Bypass (MAB) is useful for things that don't speak dot1x,like an IP camera for example. The ISE server could tell the switch to authorize a particular port if the switch knows about the MAC address of the device. When the authenticator does not get a dot1x response, it will transition to MAB. With MAB, there is **NO PERIODIC REAUTH**. This uses **RADIUS Service Type 6**.

radius-server attribute 6 on-for-login-auth

For MAB to work, it may also be necessary to add:
radius-server attribute 25 access-request include

SW1(config-if)#! Enable MAB on a switchport
SW1(config-if)#**mab**
SW1(config-if)#! Specify authentication order [1st, 2nd, …], known as **FlexAuth**
SW1(config-if)#**authentication order mab dot1x**
SW1(config-if)#! Specify priority for authentication (if both were present and available)
SW1(config-if)#**authentication priority dot1x mab**

In this example, even though mab is specified first in the order, if dot1x was later enabled on the client, it would take precedence and be used instead of MAB because of the priority command above!

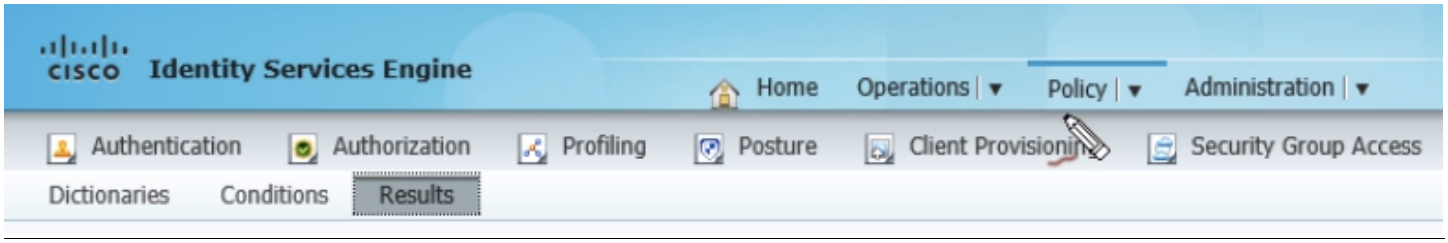
Look for "**Call Check**" in the debugs, as this indicates we are sending the MAC address over to the authentication server (ISE in this case). The username in the debugs will be the actual MAC address being passed to the server.

Look for something like "*Authentication result 'success' from 'mab' for client (xxxx.xxxx.xxxx) on Interface xxx*"

To add a MAC address for use with MAB in ISE, go to:
Administration > **Identity Management** > **Identities**. Go to **Endpoints**, then **Add**.

Go to **Operations** > **Authentications** to view the status of authentications in ISE

Identity Services Engine (ISE) + Active Directory (AD)



Make sure your name server(s) are correct in ISE before you attempt to join ISE to AD. You can enter global configuration mode just like IOS and issue ip name-server x.x.x.x x.x.x.x to specify servers. **Keep in mind that both forward and reverse DNS entries need to be present for the ISE server and the AD controllers.**

Administration > Identity Management > External Identity Sources

Choose **Active Directory**, specify **Domain Name** and **Identity Store Name** (can be anything, used to identify this AD within ISE).

Check the box next to the new Identity Store Name you have added, then choose **Test Connection**. Specify credentials. You have the option of choosing a **Basic Test** or a **Detailed Test**. The detailed test will check to ensure the necessary ports are open (LDAP/LDAPS), etc.

Once the test(s) are successful, on the same screen check the box next to the Identity Store Name and choose **Join**. Specify credentials.

Now we need to tell ISE to start using AD as an authentication source:

Administration > Identity Management > Identity Source Sequences

Click Add. Give it an intuitive name like "*Use_AD_then_Local*"… Move the Identity Store Name ("AD1", for example) from the left to the right column, repeat for "Internal Users".

Under Advanced Search List Settings, choose the radio button for "**Treat as if the user was not found and proceed to the next store in the sequence**" … the default option is "Do not access other stores in the sequence and set the "AuthenticationStatus" attribute to "ProcessError" … if left at the default, Internal Users would not be queried if AD ***TIMED OUT***… if AD failed the authentication, the buck stops there.

Now we need to enforce the Identity Source Sequence via a Policy:

Policy > Authentication

You will see defaults for **MAB** and **Dot1x**. Click the edit button next to Dot1x, choose the plus symbol, select the "*Use_AD_then_Local*" sequence we created above. Click *Save*.

Dot1x Forced Re-authentication (from privileged exec mode):

dot1x re-authenticate int gi x/x

Go to **Operations** > **Authentications** to view the status of authentications in ISE

EAP Chaining:

Policy > Policy Elements > Results > Authentication > Allowed Protocols > Default Network Access

You can uncheck any protocol you do NOT wish to allow for authentication. You can *CHECK* the box under **EAP-FAST** that says "**Enable EAP Chaining**" … then Save.

Enabling this will allow **AUTHENTICATION** of the **COMPUTER *AND* THE USER**.

ISE can leverage groups in AD and make authorization if/then decisions based upon that group membership (just like NPS).

Administration > Identity Management > External Identity Sources > Active Directory > Groups

Choose Add, then Select Groups From Directory …

Downloadable ACLs in ICE:

Policy > Policy Elements > Results

Click **Authorization > Downloadable ACLs**

Two exist by default: *DENY_ALL_TRAFFIC* and *PERMIT_ALL_TRAFFIC*

Click Add to create a new ACL. Write it as you would any other ACL.

Click **Authorization Profiles** under the same section.

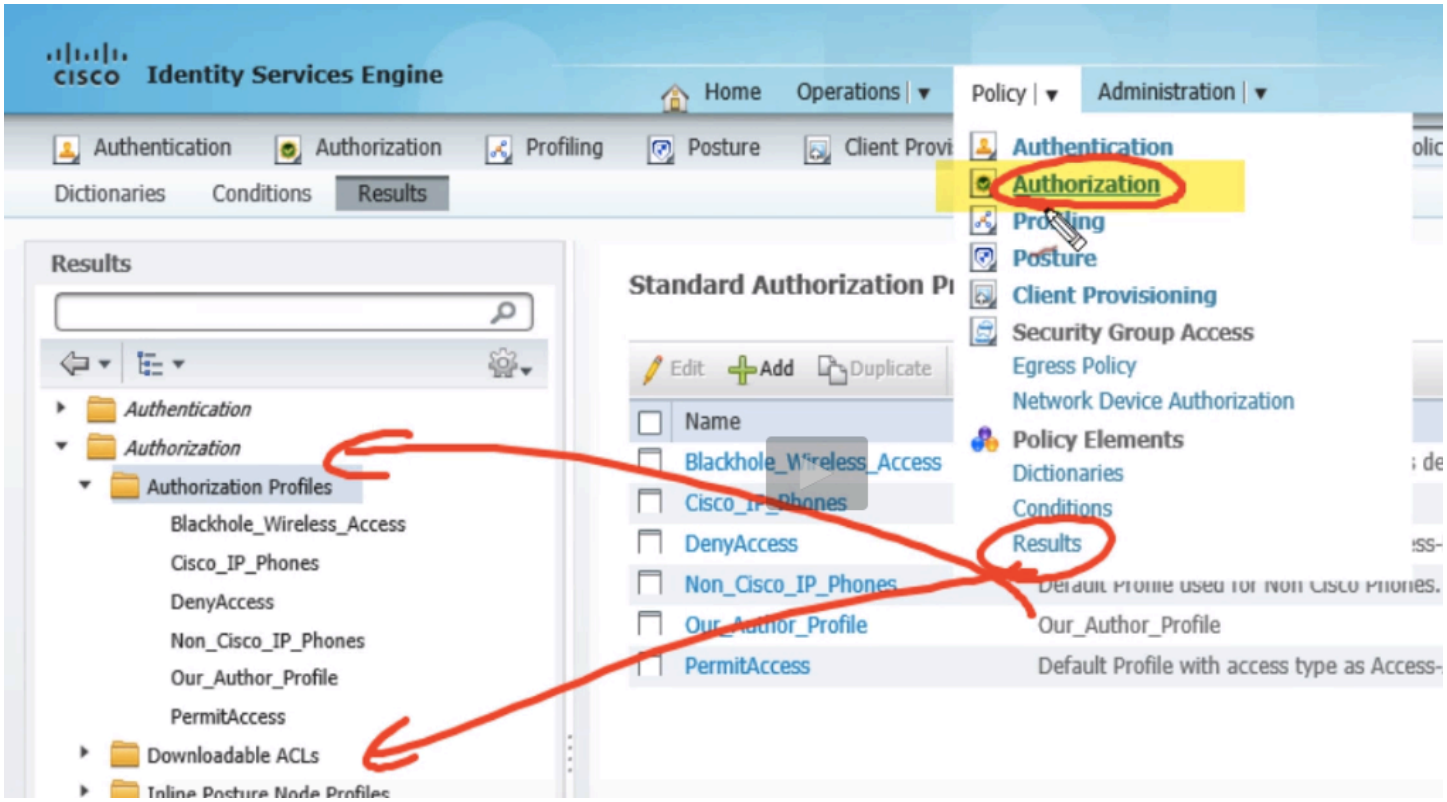
Create a new one, and then choose ACCESS_ACCEPT or ACCESS_REJECT, then check the DACL box and select the DACL PERMIT_ALL_TRAFFIC mentioned above

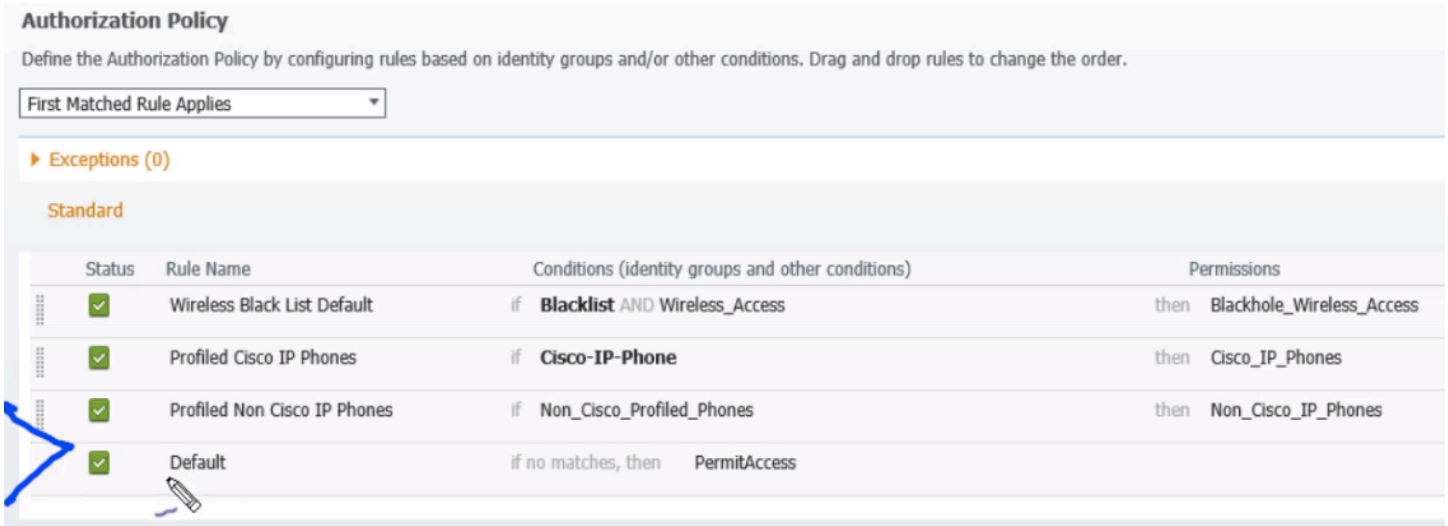
Now that we have created an Authorization Profile, we need to apply it:

Policy > Authorization *(the if/then statement section)*

You will find several defaults (Wireless Black List Default, Profiled Cisco IP Phones, Profiled Non Cisco IP Phones, Default)… **the Default at the end is a catch-all, which says if no matches then deny access.**

You could, for example, insert a new rule just above the Default rule. You could make the Condition(s) section say **if the user** *is a member of AD group x*, **then apply the aforementioned Authorization Profile**.





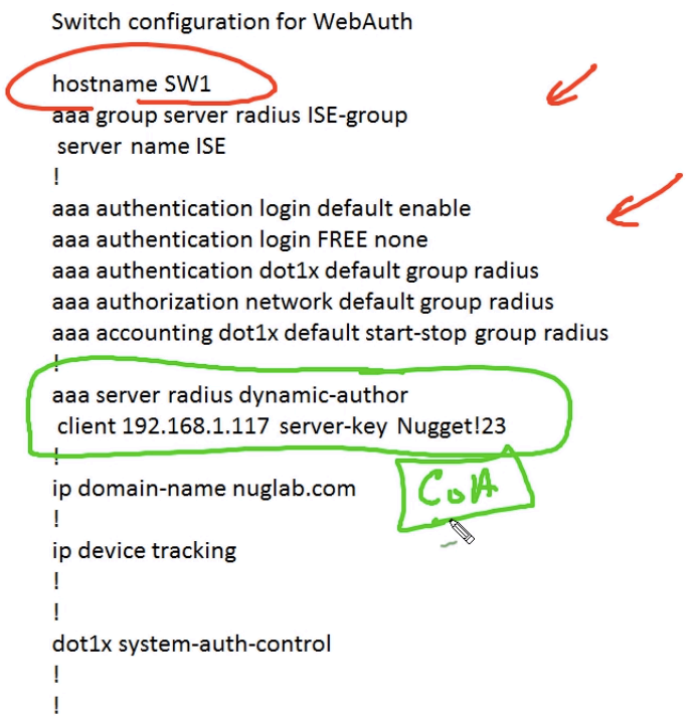
***** ProTip *****

If you want to edit the default Cisco AnyConnect wired profile (or later edit any other profile), launch the **Network Access Manager Profile Editor**, choose Networks, edit the necessary profile. When done, click Save or Save As. The default save location is **newConfigFiles**. Save your changes as **configuration.xml** and they will be applied to the locally installed AnyConnect client.

Web Authentication:

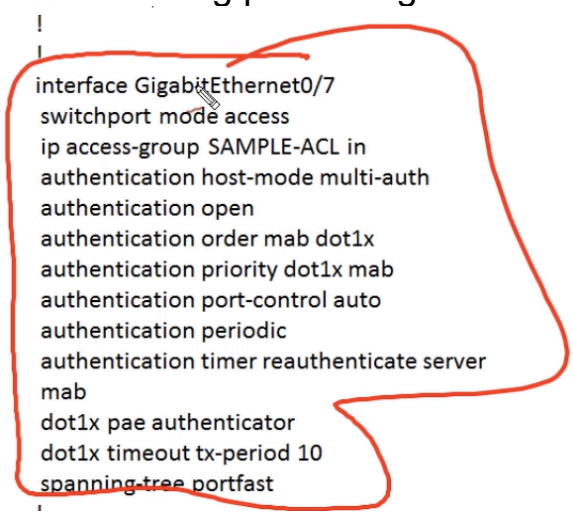
Change of Authorization (CoA / RFC 3576):
The RADIUS Change of Authorization (CoA) feature provides a mechanism to change the attributes of an authentication, authorization, and accounting (AAA) session after it is authenticated. When a policy changes for a user or user group in AAA, administrators can send the RADIUS CoA packets from the AAA server such as a Cisco Secure Access Control Server (ACS) to reinitialize authentication and apply the new policy.

The "aaa server radius dynamic-author" section below enables CoA:



If ISE wants to make a decision to de-authenticate a port, or change authorization status, this will allow the ISE server at *192.168.1.117* to reach out to the switch to make those changes. It basically tells the switch "***please be willing to accept commands as they come in from this RADIUS server…"***

The remaining port configuration is as follows:



```
!
!
interface GigabitEthernet0/7
 switchport mode access
 ip access-group SAMPLE-ACL in
 authentication host-mode multi-auth
 authentication open
 authentication order mab dot1x
 authentication priority dot1x mab
 authentication port-control auto
 authentication periodic
 authentication timer reauthenticate server
 mab
 dot1x pae authenticator
 dot1x timeout tx-period 10
 spanning-tree portfast
!
```

The access group applied to the port above can represent the controls we want in place on a port BEFORE the authentication happens. Policy can later override the ACL.

ip http server and **ip http secure-server** are required (obviously) to support WebAuth

DOT1X → MAB → Continue and redirect to web portal for authentication

We can tell ISE to try dot1x, if not try MAB, if not, continue and allow web authentication. We can invoke an ACL as below to redirect the user:

ip access-list extended REDIRECT
permit tcp any any eq www
permit tcp any any eq 443

So, MAB doesn't have the MAC in ISE, but instead of failing it continues and allows this user to be redirected to ISE where we can then present a web interface to the user to allow he/she to provide their credentials. Once authenticated, we can then push another policy down to the port to which gives the user full access.

The remaining configuration is as follows:

```
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 dead-criteria time 30 tries 3
radius-server vsa send accounting
radius-server vsa send authentication
!
radius server ISE
 address ipv4 192.168.1.117 auth-port 1812 acct-port 1813
 key Nugget!23
```

To customize the web portal in ISE (colors, logos, etc.):
Administration > Web Portal Management > Settings

Under Guest > Multi-Portal Configurations you will find **DefaultGuestPortal**. We can use that, or create our own.

You can choose the Identity Source Sequence we created earlier (AD1, for example). You can also create a **custom DACL** to **control what access is available to an individual waiting on web-based authentication**.

Policy > Policy Elements > Results

Click **Authorization > Downloadable ACLs**

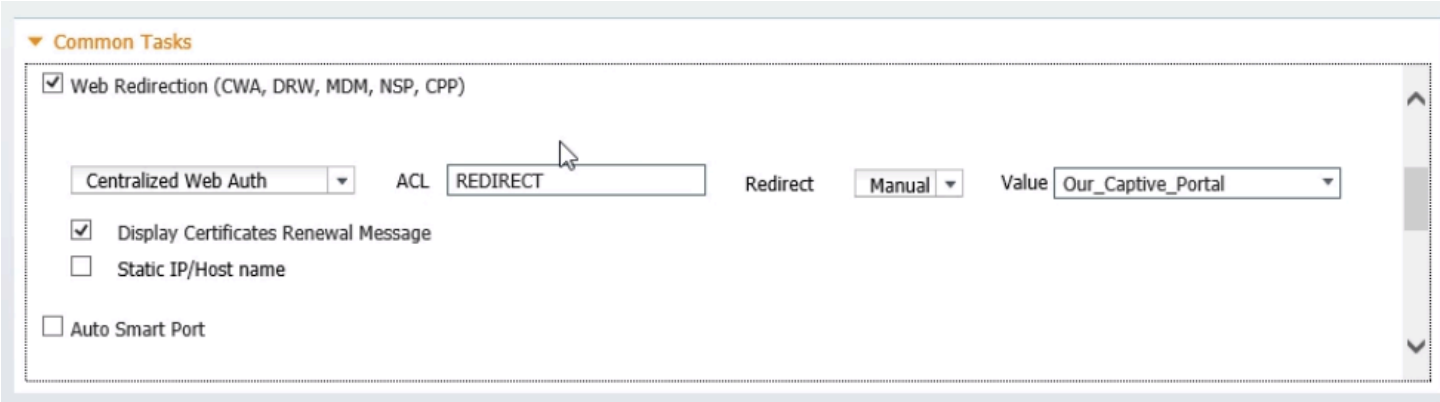
Create a new DACL by clicking Add. Call it, for example, *Waiting_for_WebAuth* … specify exactly what the user can access BEFORE authentication.

We could also create another DACL to be applied to users AFTER authentication.

Just as in IOS, creating ACLs does nothing until they are applied. Click **Authorization Profiles** under the same section.

Click Add to create a new Authorization Profile and call it, for example, WebAuth_Required. Check the **Web Redirection (CWA, DRW, MDM, NSP, CPP)** box, select **Centralized WebAuth**, then type the **name of the ACL on the SWITCH** (REDIRECT, in our example) in the ACL box. This ACL will be used to capture the traffic and redirect the user to the WebAuth portal.

In the **Redirect** box, choose **Manual** and then **select the portal**. To **use a DACL** like we created above that will be applied **BEFORE authentication occurs** (*Waiting_for_WebAuth* in the example above), check the DACL Name box and select it.



Next, create a second Authorization Profile that will be applied AFTER the user authenticates. Call it, for example, *After_AD_WebAuth*. You can optionally apply a different DACL to be used AFTER authentication.

Next, go to:

Policy > Authentication

You will see defaults for **MAB** and **Dot1x**. Edit the **MAB** section.

Change "**If user not found**" from **Reject → Continue**.

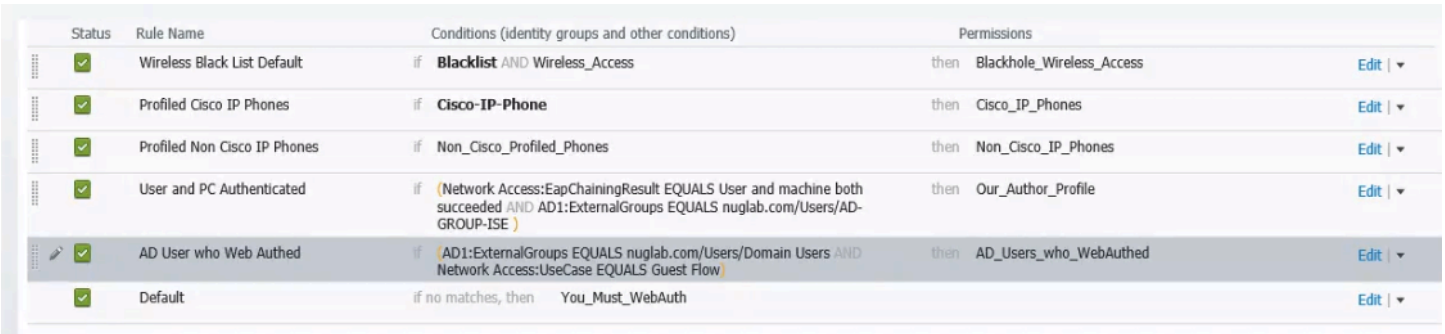
This will tell MAB not to give up, and allow us to use WebAuth!

Next, go to **Policy > Authorization *(the if/then statement section)***

Insert a new rule ABOVE the DEFAULT. Call it *WebAuth Required*, for example. Select the conditions (if) and the action (then). You could also not create a new rule and simply **edit the default rule**, sending all users not matching any of the other profiles to the WebAuth portal.

So now, let's assume user Bob is connected to a switchport. He has NO supplicant so 802.1x will FAIL, and for MAB we have not configured ISE with the MAC address of Bob's computer. Normally, the authentication would fail and it stops there. Except, we told ISE to "continue" if the MAB user was not found. It would hit the default rule (or the new rule you created) and then redirect the user to the WebAuth portal.

Now we need to create a new Authorization Policy rule to determine what happens AFTER the user has successfully authenticated via WebAuth.



Notice the Network Access Use Case is listed as **Guest Flow**, meaning WebAuth.

As always, **Operations > Authentications** will show you how things are going …

This is what it looks like when things go right. 802.1x failed, MAB failed but was told to continue …

```
SW1(config-if)#
%LINK-3-UPDOWN: Interface GigabitEthernet0/8, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface GigabitEthernet0/8, char
SW1(config-if)#do show authen session int gig 0/8
            Interface:  GigabitEthernet0/8
          MAC Address:  c8bc.c897.005c
           IP Address:  10.10.0.51
            User-Name:  C8-BC-C8-97-00-5C
               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:  N/A
              ACS ACL:  xACSACLx-IP-Waiting_for_WebAuth-5436e367
     URL Redirect ACL:  REDIRECT
         URL Redirect:  https://ISE.nuglab.com:8443/guestportal/gateway?
8B6FB8&portal=Our_Captive_Portal&action=cwa
      Session timeout:  N/A
         Idle timeout:  N/A
    Common Session ID:  010203040000001B008B6FB8
      Acct Session ID:  0x0000001E
               Handle:  0x3D00001C

Runnable methods list:
```

Notice the Redirect ACL and URL listed above. Now if we went to Bob's computer we would be redirected to the portal. Once authenticated, a new ACL will be pushed down to the port based upon the policy…

```
SW1(config-if)#do show authen session int gig 0/8
            Interface:  GigabitEthernet0/8
          MAC Address:  c8bc.c897.005c
           IP Address:  10.10.0.51
            User-Name:  it-bob
               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:  N/A
              ACS ACL:  xACSACLx-IP-AD_Users_via_WebAuth-5436e3b1
      Session timeout:  N/A
         Idle timeout:  N/A
    Common Session ID:  010203040000001B008B6FB8
      Acct Session ID:  0x0000001E
               Handle:  0x3D00001C

Runnable methods list:
       Method   State
       mab      Authc Success
       dot1x    Not run

SW1(config-if)#
```

Posture Assessment and Remediation (CPP):

Does the computer have the proper OS updates installed?
Does the computer have anti-x software installed and up-to-date?
Does the computer have a particular application installed or service running?
Does the computer have a specific registry key/value?
…

We can check Windows and OS X as well!

We do this with **Network Admission Control (NAC),** and specifically with a **NAC agent** running on the client. That data is reported back to the authentication server (ISE, in our examples).

If we have a BYOD situation, we would likely not have a full client installed but might use some kind of Java alternative.

We are going to add a **Posture Check** to the **Authorization Profile**

Posture status includes **Compliant, Non Compliant, and Unknown**

We could create 3 new authorization profiles:

1. If posture is unknown, we could redirect the user to ISE to download the NAC agent
2. If posture is non compliant, we could be placed into a **quarantine or remediation Vlan.** That Vlan should provide the user with access to what he/she needs to "get up to speed" and obtain the required software, updates, etc.
3. If posture is compliant, we could place them in the correct Vlan

Obtaining NAC provisioning resources and agent software:

Installation of a NAC agent requires administrative rights. Could push out using GPO, or you could use the ISE server itself to deploy the software.

To prepare ISE server to download the NAC software and provisioning resources (Rules, AV / AS version info, etc.), go to:

Administration > System > Settings

Click **Posture > Updates**

There is a built-in URL to download **Posture Updates**. Obviously ISE server needs DNS resolution to resolve that URL. Can set automatic updates here as well.

To obtain the actual **NAC software**, go to:

Policy > Policy Elements > Results

Click **Client Provisioning > Resources**

By default, you will see nothing here. Click **Add**, then choose "**Agent resources from Cisco site**" … a list will be populated showing the available resources. Download the necessary agents that the ISE server will use in provisioning the client.

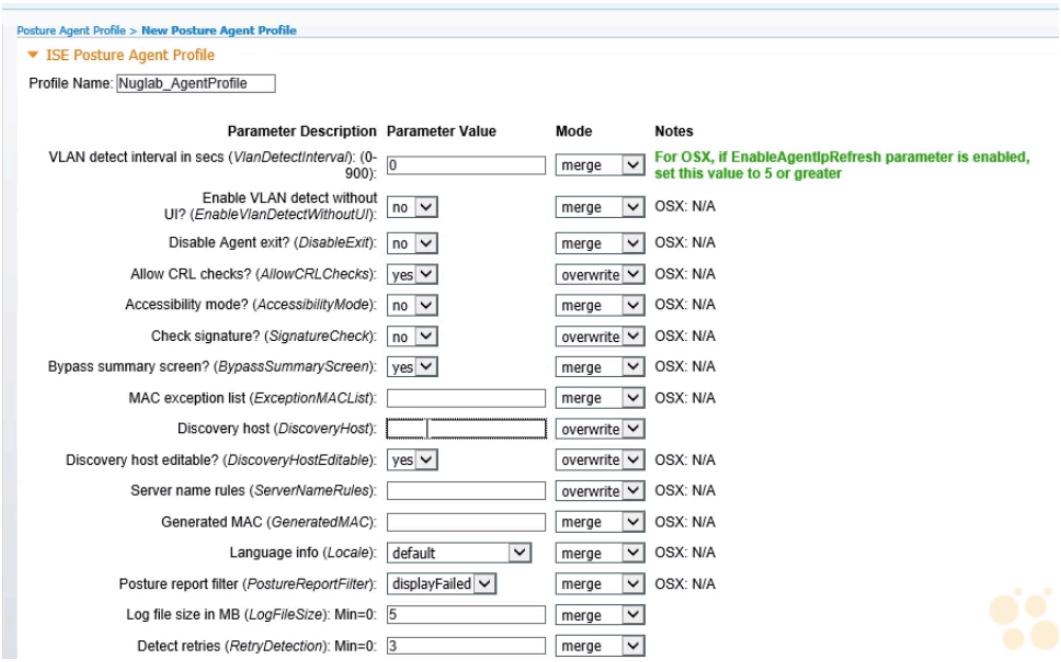
Provisioning NAC agents from ISE:

The first step is to create a Posture Agent Profile:

Policy > Policy Elements > Results

Click **Client Provisioning > Resources**

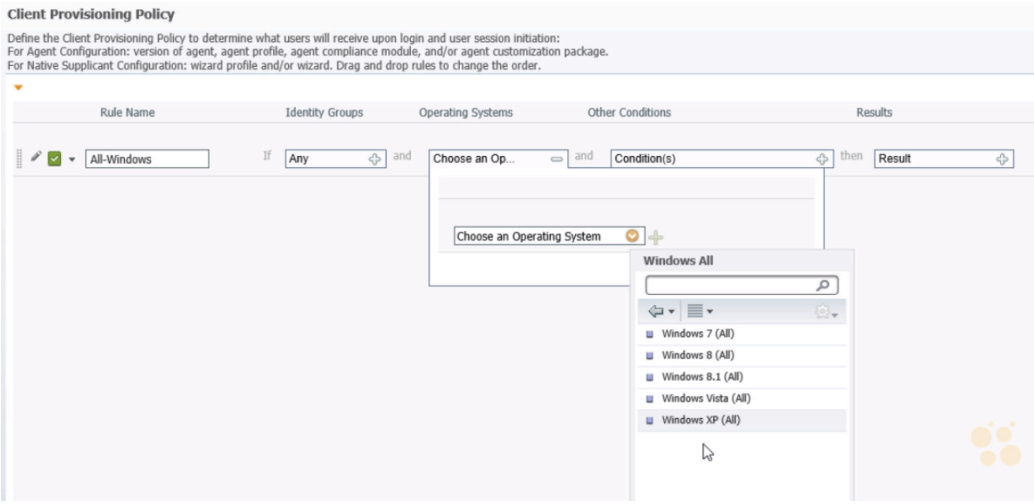
Click **Add**, then choose "**ISE Posture Agent Profile**". There are many things that can be customized here, most specifically the **Profile Name** and the **Discovery Host**. This is the location to which the agent will report (the ISE server).



Next we need to create a **Client Provisioning Policy**:

Policy > Client Provisioning

We can specify certain versions of Windows that require an agent, or choose Windows All to include all versions.



We can create certain conditions, such as only apply the policy if the user is a member of AD, or of a certain AD group. You can specify the agent profile and a compliance module.

Next we need to create an **Authorization Profile** to redirect the user to the portal if they are not running the NAC agent:

Policy > Policy Elements > Results

Click **Authorization > Downloadable ACLs** to create a DACL that can be used in conjunction with the Authorization Profile. *This determines what the client can access while the client is obtaining its agent.*

Click **Authorization > Authorization Profiles**

Create a new one, and then choose ACCESS_ACCEPT, then check the DACL box and select the DACL mentioned above. The most important piece is the **Web Redirection (CWA, DRW, MDM, NSP, CPP)** box. Instead of selecting Centralized Web Auth as we did earlier, select **Client Provisioning (Posture)**. As we did earlier with the WebAuth portal, we can invoke an ACL as below to redirect the user:

ip access-list extended REDIRECT
permit tcp any any eq www
permit tcp any any eq 443

Next we need to apply the Authorization Profile:

Policy > Authorization *(the if/then statement section)*

You can modify an existing rule (User and PC Authenticated) to add a condition of **Session:PostureStatus Equals Compliant**. But what happens to a user who has no agent? We can duplicate the rule we just modified and make a similar rule that has the condition of **Session:PostureStatus Equals Unknown** and insert it right after the previous rule. Choose the **Authorization Profile** we created above that will **redirect the client to where they can obtain the NAC agent**.

When the client attempts to connect without a NAC agent, they will be redirected to the provisioning portal.

Profiling Endpoints:

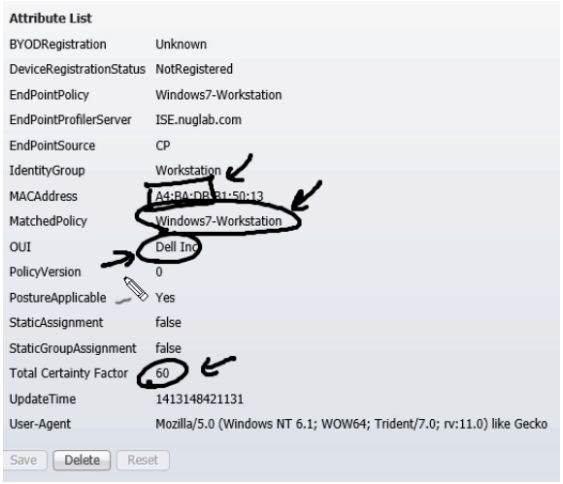
How many iOS devices are on our network? How many Android devices, or Windows 7 computers, etc.?

As devices connect to our network, we can automatically profile them and tie that to an authorization profile, which would in turn dictate the resources to which those devices had access.

Without even turning on additional features, we can already obtain some device information via:

Administration > Identity Management > Identities. Go to **Endpoints**. If you have manually added any systems for **MAB**, they will appear here. However, details discovering endpoints can also be learned **dynamically**.

For example, you may see a system listed under **Endpoint Profile** called **Windows7-Workstation**. This indicates **ISE automatically classified** the computer and put it into an **Identity Group called Workstation**. The reason is because it **matched a Policy called Windows7-Workstation**. The **confidence** ISE has that the information is correct is recorded via the **Total Certainty Factor**.



By default, ISE isn't doing as much as it can do regarding classification of endpoints. If we want to tell ISE to do serious profiling, go to:

Administration > System > Deployment

You will see the ISE server listed here. You have the option of making the server a **PRIMARY** server instead of a **STANDALONE** server. Once you do that, the **Profiling Service** will be active. Click the **Profiling Configuration** tab. Nothing is selected by default, but we can choose things like **NETFLOW, DHCP, DHCPSPAN, HTTP, RADIUS, NMAP, DNS, SNMPQUERY, SNMPTRAP**. The NMAP option is interesting because you can run a scan of a particular subnet right from this screen which will add any discovered endpoints to the Administration > Identity Management > Identities > Endpoints section described above.

If we go to **Administration > Network Resources > Network Devices**, we can add switches and other devices. Within the configuration for those devices we can turn on SNMP.

However, to set up **SNMP credentials ISE can use to poll devices on the network**, go to **Administration > System > Settings**, then click **Profiling**.

To **view or modify Profile Policies** in ISE, go to **Policy > Profiling**. There are hundreds of policies to choose from, and they can be sorted and filtered in the Show dropdown menu. You can edit the policies and specify NMAP actions and other conditions which can be used to **increase or decrease the Certainty Factor** mentioned above.

Administration > Licensing is a key area to view the number of used and remaining endpoints the license covers.

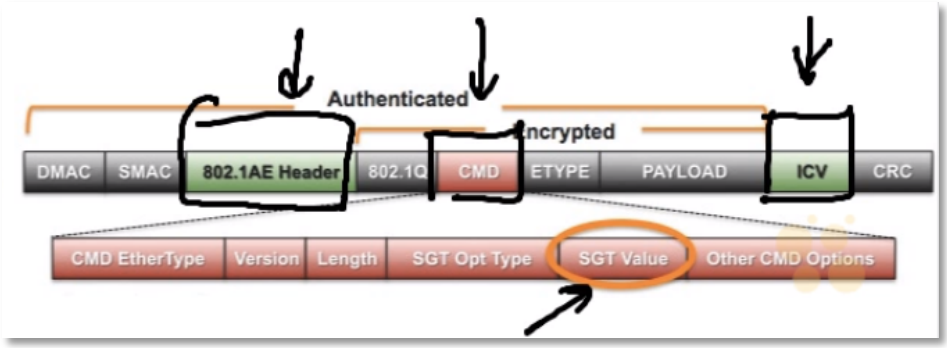
MACsec and TrustSec:

MACsec is data encryption and security at layer 2. It uses a **different frame format** than traditional Ethernet.

- **Confidentiality** via AES
- **Integrity** via Integrity Check Value (ICV)

Host to Switch → MACsec Key Agreement (MKA)
Switch to Switch → Security Association Protocol (SAP, Cisco)

New 802.1ae header, CMD field for vendor-specific info, and ICV field



MACsec can be extended further than just a host and the switch to which it's connected. For example, the **switch-to-switch path** could be protected as well via **SAP**. The important thing to remember is that it is **HOP-BY-HOP ENCRYPTION**, **NOT END-TO-END** like IPsec.

MACsec between two switches could be manually configured, or dynamically negotiated via ISE.

Cisco extended 802.1ae with **TrustSec** via the **Security Group Tag (SGT) value** (within the **CMD** field in the Ethernet header).

For example, we could have **SGTs representing groups** like sales, engineering, marketing, etc. We could assign tags (numbers) to each respectively – 1, 2, 3.

Example:

- Deny frames with SGT 1 (sales) if they are using specific protocols destined for SGT 3 (marketing)
- Permit frames with SGT 2 (engineering)

Rules can be implemented as **Security Group ACLs (SGACL)**. These rules could be centrally managed and pushed down from ISE or ACS.

The problem is, **not every device will be TrustSec capable**. To address that, we can use **SGT Exchange Protocol (SXP)**. It's like DNS for SGTs. For example, we could have ISE **inform our devices that can't process the labels** of the **IP address to SGT mappings**. Now the device can still implement SGACLs based on the tags due to the learned mappings.

Centrally Implementing TrustSec:

In order to centrally implement TrustSec, there are a few things that must be done in ISE, and a things that must be done on the Network Access Device (NAD) (the switch).

ISE:

- SGA password for NAD
- SGT groups
- SGACLs (create and apply)

NAD:

- AAA method list for CTS
- CTS and PAC credentials for NAD
- Enforce role-based (SGACLs)

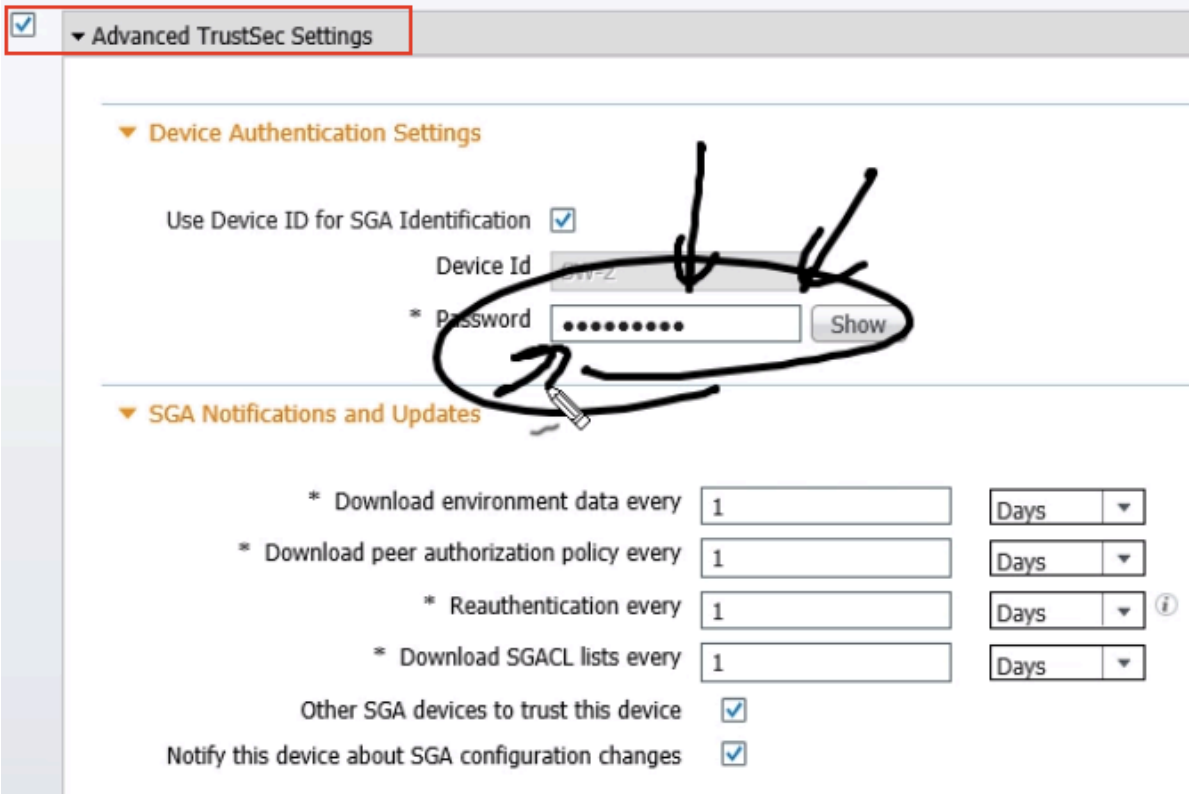
Go to:

Administration > Network Resources > SGA AAA Servers

… by default, the ISE server is already on this list. We have to tell ISE that our NAD is going to be using SGA:

Administration > Network Resources > Network Devices

Check the Advanced TrustSec Settings box and enter the Password to be used between ISE and the NAD (the switch):



Also check Notify this device about SGA configuration changes

You may also want to configure the **Device Configuration Deployment** section and enter the credentials to **allow ISE to push SGACLs to the switch**.

Now we need to **create Security Groups**:

Policy > Policy Elements > Results

Click **Security Group Access > Security Groups**

By default, we will have one security group called Unknown. As you create additional groups, ISE will automatically assign a Security Group Tag (SGT).

Next, under the same Security Group Access tree, click **Security Group ACLs**:

When you create an SGACL you will notice there is **no source or destination mentioned.** We can say things like "*deny icmp*", "*permit ip*", etc. The reason why the source/destination is omitted is because we can choose to apply the SGACL between two different security groups. These can be dynamically applied based upon where the traffic is coming from / going to. This is also known as **Role-based ACL (RbACL).**

Lastly, under the same Security Group Access tree, click **Security Group Mappings**:

These mappings can be used to **manually associate an IP address with an SGT** (*example: if the IP address is 8.8.8.8 it should be associated with the security group called SERVERS. The security group SERVERS could then have an associated SGT of "3"*).

How do we associate specific Security Groups with users?

Policy > Authorization *(the if/then statement section)*

In the "then" section of the if/then lines, you could associate specific conditions with the application of a specific Security Group, which in turn is associated with a specific SGT.

NAD Configuration:

Next, we must proceed to the NAD (the switch) to complete that side of the configuration. The commands are as follows:

adding for TrustSec:

```
conf t
radius-server host 192.168.1.117 pac key Nugget!23
aaa authorization network cts-author-list group radius
cts authorization list cts-author-list
exit

cts credentials id SW-2 password Nugget!23

show cts credentials

show cts pacs

show cts environment-data

int gig 1/0/12
cts role-baled enforcement
end

cts refresh environment-data

show cts environment-data
```

The first line, **radius-server host *x.x.x.x* pac key *xxx*** specifies the **Protected Access Credential (PAC)**.

The **Cisco TrustSec (CTS)** line, **cts authorization list *xxx*** references the **aaa authorization network *xxx* group radius** command just above it.

The **cts credentials id *xxx* password *xxx*** supplies the TrustSec credentials for this NAD. This key must match the password configured above in the Advanced TrustSec Settings of Network Devices.

The interface command **cts role-based enforcement** specifies the ports on which we want to enforce TrustSec.

The command **cts refresh environment-data** will **refresh/update the information from the ISE server**. The **show cts environment-data** shows the aforementioned data.

The command **show cts credentials** will show the device ID and whether credentials are configured on the switch.

```
SW2#show cts credentials
CTS password is defined in keystore, device-id = SW-2
```

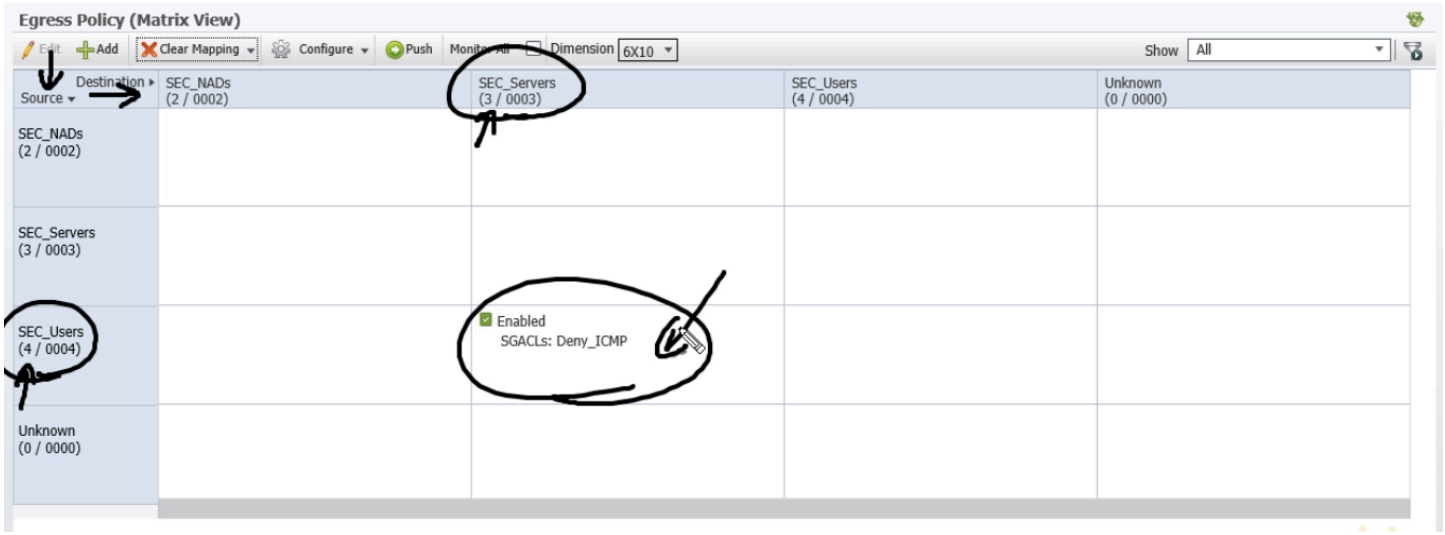
The command **show cts pacs** will show information about the Protected Access Credentials, and **show cts environment data** will show the **Security Groups learned from the ISE server**.

We need to tie it all together via the application of SGACLs.

Policy > Policy Elements > Results

Click **Security Group Access > Matrix**

Find the box at which the source/destination tags intersect and double-click to apply the SGACL (RbACL).



The interface command **cts role-based enforcement** will insure the switch is actually enforcing the ACLs. *This command WILL VARY depending on the IOS version and hardware platform.*

Verification via show authentication session int *x/x*:

```
SW2#show authen session int gig 1/0/7
            Interface:  GigabitEthernet1/0/7
          MAC Address:  a4ba.dbb1.5013
           IP Address:  192.168.1.120
            User-Name:  it-bob
               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:  1
              ACS ACL:  xACSACLx-IP-PERMIT_ALL_TRAFFIC-537cb1d6
                 SGT:  0004-0
      Session timeout:  N/A
         Idle timeout:  N/A
    Common Session ID:  C0A8017E000000200082BB6E
      Acct Session ID:  0x0000002B
               Handle:  0x6E000021

Runnable methods list:
       Method   State
       mab      Not run
       dot1x    Authc Success
```

ISE Personas:

Addresses the need for fault tolerance in ISE. An ISE server could serve role as **PRIMARY**, **SECONDARY**, or **STANDALONE** ISE server.

The services provided by **Personas** include:

- **Administration** (system configuration and settings)
- **Policy Service** (referred to as a **Policy Service Node – PSN**)
- **Monitoring**
- **Inline Posture** (provides **Network Admission Control – NAC**)

These roles can be spread across multiple ISE servers. NTP is critical across all of those systems.

Sponsor Portal and Guest Access:

We can manage guest access using ISE (similar to Prime Infrastructure).

ISE uses the concept of Portals:

- **Admin** – the main portal where an IT or IS user would access ISE (in a **distributed environment**, this would be the server acting as the **Administrative Node**)
- **Sponsor** – this portal is used by the **Lobby Ambassador** to allow non-IT people to manage guest access to the wired or wireless network (in a **distributed environment**, this would be the server acting as the **Policy Service Node**)

The credentials created in the Sponsor portal can be printed, emailed, or SMS/texted. The credentials would then be entered by the guest user via the **Centralized Web Authentication (CWA)** web page.

Once the guest has authenticated, an **Authorization Profile** can be applied to give the guest user the specific access determined by the company.

In the event we don't have a person serving as Lobby Ambassador, we can implement **Self-Registration**. The obvious downside to this is that without a human being serving as the LA, we cannot verify the identity of the person requesting access.

Implementing the Sponsor Portal:

We need to enable SMTP services on ISE so it can send email (or an email gateway to send text messages).

ISE comes with a preconfigured Guest Identity Management Group:

Administration > Identity Management > Groups

Of the defaults listed, you will see **Guest**. We can create additional groups here as well.

To configure an SMTP server:

Administration > System > Settings

Click **SMTP Server** and enter the proper settings.

To configure the port on which the Sponsor Portal is served, go to:

Administration > Web Portal Management > Settings

Click **General > Ports**. You will notice the default is **8443**.

Under the same **Settings** section, click **Sponsor > Language Template** and you can customize the contents of the email messages, text messages, etc. *per language*.

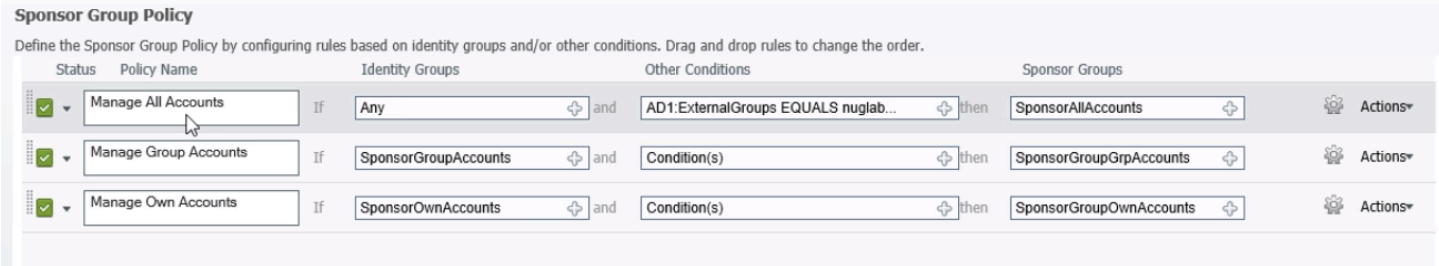
Under the same **Settings** section, click **Guest > Details Policy** and you can customize the required fields necessary for creating guest accounts (first name, last name, company, email, phone, optional data, etc.).

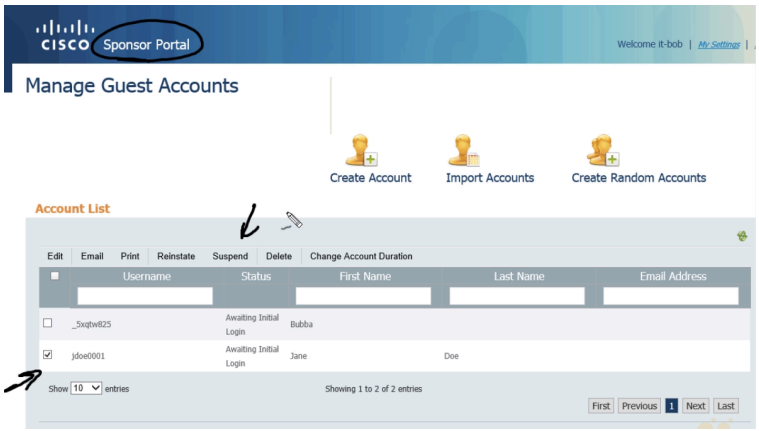
Under the same **Settings** section, click **Sponsor > Authentication Source** and you can specify the **Identity Source Sequence** to be used for access to the Sponsor Portal.

To configure who can **access the Sponsor Portal**:

Administration > Web Portal Management > Sponsor Group Policy

You can specify if the user is a member of certain AD groups, he/she can access the portal.





BYOD:

Example steps for BYOD implementation:

1. Redirect all unknown devices to a WebAuth portal to authenticate the user.
2. Authenticate the user.
3. Redirect the user for provisioning.
4. ISE does **identity certificate enrollment** and wizard for installation on device.
5. 802.1x configuration wizard.
6. 802.1x authentication.

ADDITIONAL NOTES FROM IPEXPERT TRAINING:

TrustSec has 3 phases of deployment:

- Monitor
- Low Impact
- Closed

RADIUS Attribute #26 (0x1A) is called Vendor-Specific

Cisco uses VSA **Cisco-AV-Pair (Vendor-ID 9, Vendor Type 1)** to extend TACACS+ authorization capabilities to RADIUS. Cisco-VPN-3000 is another attribute commonly used.

Use "aaa authorization config-commands" to make Global Configuration commands subject to authorization (could be used with TACACS+ and ACS to provide granular command authorization)

For WLCs, enable AAA Override to allow the RADIUS server to override any configured security policies on the WLAN. RADIUS NAC is also enabled to assist with this.

RADIUS **non-EAP** authentication methods include:
PAP, CHAP, MSCHAPv1, MSCHAPv2
… AAA server returns an Access-Accept or Access-Reject message

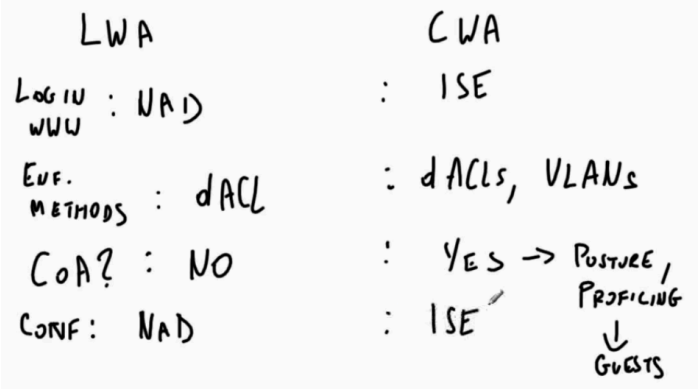
RADIUS **EAP-based** authentication **key-based** methods include:
EAP-MD5 (weak, one-way client authen), LEAP (better than EAP-MD5 but still not good, mutual authen, wireless ONLY), EAP-FAST (uses PACs as a shared secret, inner tunnel is EAP-Generic Token Card (GTC) or EAP-MSCHAPv2)

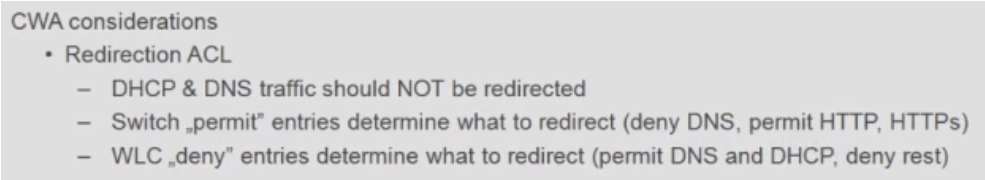
RADIUS **EAP-based** authentication **certificate-based** methods include:
PEAP, EAP-TLS (most secure, client has certs as well as server, could even use smartcard to store client's pub/private keys)

The process is:

1. Host connects to network device (switch or WLC)
2. Network device sends **EAP Request** to host
3. Host replies with **EAP Response**
4. Network device **encapsulates EAP Response** from host **into RADIUS Access-Request** and **sends to ISE**
5. **Cisco ISE extracts EAP Response** from RADIUS packet and creates new **EAP Request**, encapsulates it into **RADIUS Access-Challenge**, sends to network device
6. Network device extracts **EAP Request** and sends to host

Local WebAuth (LWA) vs. Central WebAuth (CWA):





Sponsor Portal URL:
https://ise_ip:8443/sponsor

Guest Portal URL:
https://ise_ip:8443/guest

We can enable multiple protocols for profiling:
DHCP, DNS, HTTP, RADIUS, etc.

RADIUS is the simplest because the NADs are already talking to ISE. We can leverage this by carrying some additional information about the connecting devices so we can profile them. When a device is connected to the NAD, the NAD will send a RADIUS **Access-Request** message to ISE. This packet will contain the addresses of the device connecting (**MAC, IP, or both**), and the remaining part are **Profiling Attributes** (technically, the MAC and IP are attributes as well).

The info gathered will be stored in the Endpoint Database, which is built upon MAC addresses as the identifier of the endpoint.

If it doesn't already exist, a new entry will be added to the Endpoint Database. From the OUI, ISE can determine the vendor of the NIC.

We could also use **HTTP Probes**. If the endpoint connects to CWA to authenticate, we can leverage the **User Agent** string and determine the **OS** and other info. The problem is that the Endpoint Database uses MAC addresses as identifiers, not IPs, yet we can only obtain an IP from the HTTP Probe. **The information would be discarded because there is no way to correlate the data.**

Now, if we were capturing **DHCP** data, we would know the **MAC > IP mapping** an could use that info and update the database accordingly.

Go to **Policy > Profiling** to edit the policies that determine how the attributes are used and collected. For example, there will be an entry for Apple-Device that contains child policies for MacBooks, iPhones, iPads, etc.

The **Minimum CF (Certainty Factor)** can be modified under the profiles. The conditions within the profile can increase the CF. If the conditions increase the CF to at least the minimum, the device will be assigned to that profile.

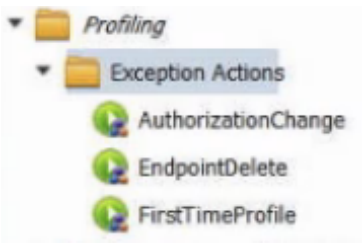
If we had policies with overlapping conditions, they will ALSO be checked. The winning policy depends on the total CF value.

For example, Apple-Device has a minimum CF of 10, and a condition increases the CF by 10 if the OUI is Apple. That makes the total 10. Under the Apple-Device policy, a child policy called Apple-MacBook exists. It checks the user-agent string for "Macintosh" and "Mac OS". If a match is found, it increases the CF by 20, making the total CF 30. The new policy matched will then be Apple-MacBook.

The entire purpose of using these policies is to match an Identity Group. We have the option within the policy of "**Create Matching Identity Group**" or "**Use Hierarchy**". In our Apple-MacBook example, we would choose "Use Hierarchy" and then look at the Parent Policy, Apple-Device. The Parent Policy here will be ***NONE***, meaning that an Identity Group of the same name will be matched. If we chose the create group option instead, a new group would be created as "Apple-MacBook" instead of the parent. ***It depends on how granular you want to get. Do we want a separate Identity Group for each sub-Apple-device, or will the parent device suffice?***

We have the option of adding an NMAP scan action under these profiles.

- CoA (RFC 3576) is an unsolicited RADIUS message sent to NAD to enforce a new policy
  - This process is triggered automatically (if enabled globally) under one of conditions below :
    1. Endpoint is added/removed from an identity group that is used by an authorization policy
    2. Endpoint is profiled for the first time
    3. Endpoint is deleted from the ISE database



The Exception Actions above exist by default. We can create new actions to trigger CoA events as well.

We can also manually create an endpoint and statically assign a policy by modifying the Endpoint Database.

Probes:

Some probes are only useful if an IP-MAC binding already exists (as mentioned above).

Probe Types (all are *PASSIVE* except NMAP)

RADIUS Probe

- MAC/IP addresses
- Provides IP to MAC bindings (Framed-IP, Calling-Station-ID)
- Can be extended by enabling Device Sensor feature

A **Device Sensor** is an **extension** to the RADIUS Probe. Enables NAD (switch or WLC) to collect info through **CDP, LLDP, and DHCP**. Sent to ISE via RADIUS Accounting Packet. In the WLC, go to WLAN > WLAN ID > Security > AAA Servers. Activation via WLAN > WLAN ID > Advanced > **Device Profiling**. Supports DHCP Proxy and Bridged modes. Make sure to enable "vsa send accounting" on the switches and "device-sensor accounting" and "device-sensor notify all-changes".

SNMP Trap Probe

- Typically used to trigger SNMP Query Probe
- MAC address can be collected if MAC Notifications enabled for port
- Traps from WLCs and APs NOT supported
- Not much info here – "*Hey! This port just went up/down. Something happened! If you want to know more, send an SNMP Query (GET)!*"
- **Redundant info if you use RADIUS Device Sensor**

SNMP Query Probe (GET)

- Periodic or triggered on reception of SNMP Trap/RADIUS Accounting message
- Key profiling attributes: CDP/LLDP, ARP Table
- Provides IP to MAC bindings (ARP Cache)
- **Redundant info if you use RADIUS Device Sensor**

NetFlow Probe

- Used to identify endpoints based on the traffic they generate
- *IP to MAC binding must already be known by ISE!*

DHCP & DHCP SPAN Probe

- Use SPAN Probe if there is no Relay configured
- Contains DHCP packet info
- Provides IP to MAC bindings
- Could use ip helper-address to send traffic towards ISE

HTTP Probe

- Use SPAN Probe if URL Redirection or Client Provisioning not available
- **User-Agent** string is key info provided by this probe
- HTTP traffic does NOT include MAC, so binding must exist in ISE

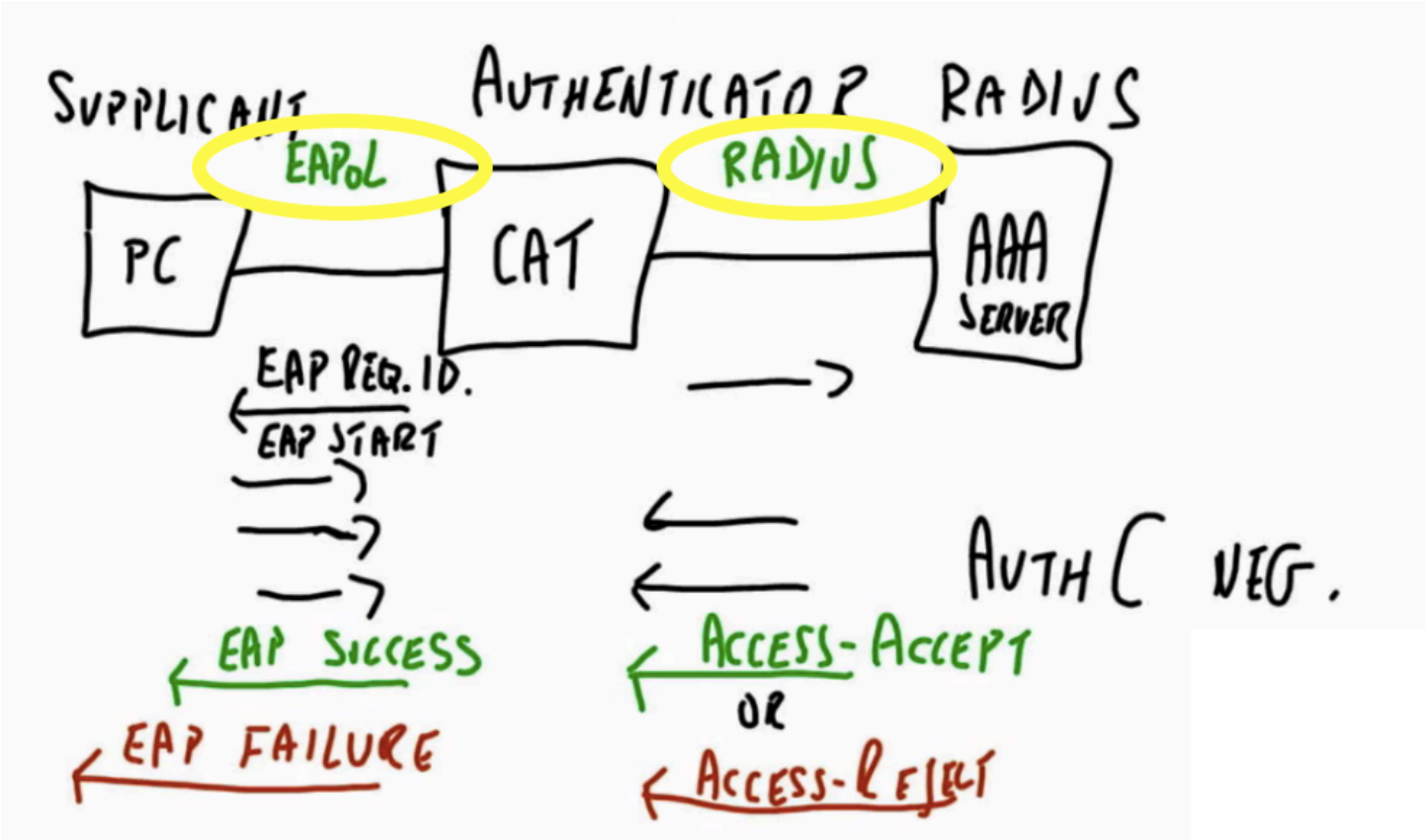
DNS Probe

- Used to acquire FQDN based on rDNS lookup
- IP address of endpoint must already be known by ISE (obviously)
- Probably not very useful

NMAP Probe

- The only ACTIVE probe type
- **OS, SNMP, Common Ports**
- Scans started manually or dynamically by Policy
- IP address of endpoint must already be known by ISE (obviously)

802.1x Authentication Process:



For **Dot1x Port Violations**, we have the same 3 options as Port Security, which work the same way:

- **Protect**
- **Restrict**
- **Shutdown**

… plus a new 4th option:

- **Replace** (replaces the currently authenticated MAC with the new one)

Set with the **authentication violation *x*** command

These violations can ONLY be triggered with **single-host** or **multi-domain** modes

We also have **MAC Move**:
When enabled, re-connecting (moving) an already authenticated device to another port will trigger **re-authentication instead of a violation**

To enable, use **authentication mac-move permit**

Default Dot1x Timeout: 3 x 30 seconds (90 seconds)
To adjust, use **dot1x timeout tx-period *x* (change to 10 seconds as best practice)**

MAB **Wired** (Ethernet), Service-Type **10 (Call-Check)**, NAS-Port-Type **15**
MAB **Wireless**, Service-Type **10 (Call-Check)**, NAS-Port-Type **19**

Guest VLANs:
Compatible with MAB, assigned to clients without supplicant installed (for limited access), not for multi-auth ports:

Configure with **authentication event no-response action authorize vlan *x***

Auth-Fail (Restricted) VLANs:
Used for clients that FAIL Dot1x authentication, NOT compatible with MAB or WebAuth, if configured, any other fallback method will NOT be used, only for single-host ports

Configure with **authentication event fail action authorize vlan *x***

Critical VLAN (for failed AAA servers):

Configure with **authentication event server dead action authorize vlan *x***

Low Impact Mode Configuration (ISE)

- Profiling should be already enabled
- Authentication Policy Default Rule can be set to „Deny" (we only want MAB or 802.1x)
- Define dACLs
- Appropriate RBAC/Device Authorization Profiles should be now tuned :
  - Specify dACLs and/or VLANs (number or name)
  - Also create a profile for CWA
- Authorization Rules should be created/tuned as needed
  - The Default AuthZ Rule should point to Central Web Authentication

Closed Mode (wired & wireless networks)

- No traffic (except for EAPOL/STP/CDP) flows through the port priort to successful authentication
  - Perfect mode for VLAN assignment
- VLAN assignment or dACLs enforce the policy
  - Make sure all assignable VLANs are defined on every switch
    - On WLC interfaces must be defined that correspond to the required VLANs
  - If a non-existing VLAN is attempted to be assigned, authorization fails
  - Avoid using multi-auth mode – only the first assigned DATA VLAN will be used
- Auth-Fail (Restricted) VLAN can be configured to be assigned to users who failed 802.1x

Closed mode is the ONLY mode for wireless. It's 0 or 1, pass or fail authc.

On a WLC, if you want to enable guest access on a WLAN via WebAuth, turn Layer 2 security OFF, and turn ON MAC Filtering.

Turn ON AAA Override, change the NAC state to RADIUS NAC

Vlan DHCP Release can be turned on under the Guest Portal to enable an applet download that will re-request an IP address on a guest computer after the guest authentications via the Web Portal.

Wireless Dot1x

Unlike Wired Dot1x, the DACLs for WLCs are defined on the WLC itself, NOT in ISE. When you create the Authorization Policy in ISE, you will choose **Airespace ACL Name** instead of **DACL Name**.

Also, on WLCs, the access list to match the REDIRECT traffic is **opposite** as it is on a switch. Instead of using permit statements, use **DENY to MATCH the traffic you want to redirect on the WLC.**

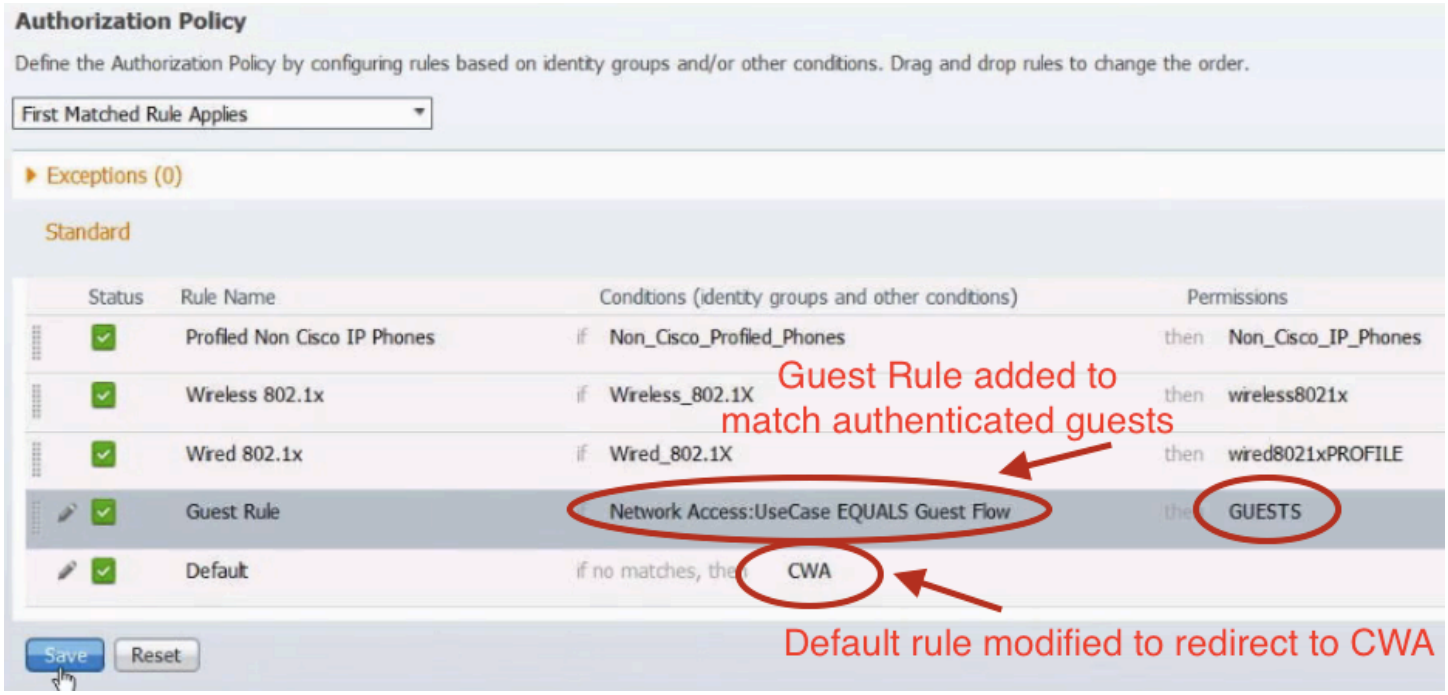
You would want to create an ACL that PERMITs traffic from the WLC to ISE, as you do NOT want to redirect that traffic. You would also NOT want to redirect DHCP and DNS packets.

So, an example ACL would be:

Seq	Action	SIP	DIP	Pro	SP	DP	Direction
1	Permit	Any	*ISE_IP*	Any	Any	Any	Inbound
2	Permit	Any	Any	UDP	DHCP_C	DHCP_S	Inbound
3	Permit	Any	Any	UDP	Any	DNS	Inbound
4	Permit	Any	Any	Any	Any	Any	Outbound
5	*Deny*	*Any*	*Any*	*Any*	*Any*	*Any*	*Both*

Sequence 4 is permitting the return traffic.
Sequence 5 is the **implicit deny**, and that **DENY** would define the **REDIRECT** traffic.

Example of Authorization Policy to enable CWA and Guest Access:



MACSec:

Follows the regular Dot1x authentication process. Uses 4 encryption settings:
Should-secure, Must-secure, Must-not-secure, not-MACsec-capable

Configuration **(in addition to Dot1x):**

interface *interface*
mka [policy_name | default-policy]
authentication linksec policy [should-secure | must-secure | must-not-secure]
macsec

While you *can* manually configure this as above, this is most commonly deployed centrally via ISE. **The config applied to the switchport can serve as a fallback if ISE is unreachable, but the ISE settings will always OVERRIDE the locally defined settings.**

MACsec will only be applied in the following policy combinations:
Should-secure + Should-secure
Must-secure + Must-secure
Must-secure + Should-secure

The supplicant and the switch must support Dot1x + MACsec

Switch-to-Switch Mode:

Manual Mode (NO Dot1x)
Dynamic Mode (REQUIRES Dot1x)

Configuration **(Manual):**

interface *interface*
cts manual
sap pmk *key-in-hex* **mode-list** [gcm-encrypt | gmac | null | no-encap]*****
no propagate sgt

gcm-encrypt	=	auth + enc
gmac	=	auth, no enc
no-encap	=	no encapsulation (MACsec is **DISABLED**)
null	=	encapsulation, no auth, no enc

***You can stack multiple modes as preferred and the method will be negotiated**

show cts interface *x/x* to verify
Look for SAP Status: SUCCEEDED
"**Selected cipher**" will show which cipher was negotiated from the list

If the device on the other side does NOT support MACsec, you will see "**sap fail**" increment under statistics.

You can edit/create an **Authorization Profile** in ISE and choose **MACsec Policy**, followed by **Should-secure, Must-secure, Must-not-secure**

MACsec Switchport Configuation:
mka default-policy
macsec

You will need to configure “802.1x (MACsec)” in the supplicant (AnyConnect)

show macsec summary will provide information on the MACsec-enabled interfaces
show macsec interface *x/x* will provide detailed information

Final switchport configuration:

```
interface GigabitEthernet1/0/2
 switchport mode access
 switchport voice vlan 808
 ip access-group PREAUTH in
 authentication event fail action next-method
 authentication host-mode multi-domain
 authentication open
 authentication order dot1x mab
 authentication priority dot1x mab
 authentication port-control auto
 macsec
 mka default-policy
 mab
 snmp trap mac-notification change added
 dot1x pae authenticator
 dot1x timeout tx-period 10
 spanning-tree portfast
end
```

Posture Assessment:

NAC Agent is required on the endpoint to evaluate the posture status

There are two types:
Permanent – installed on endpoint
Temporary - web agent (useful for guests, but not all remediation supported)

The Dot1x authentication will happen like normal. If it is successful, the device is going to have its posture status initially set to ***Unknown***. The **Authorization Policy** will return the **redirection URL** and **redirection ACL** used to direct the user to the **Client Provisioning Portal** where the user can **download** the **NAC Agent**.

The NAC Agent will perform posture assessment when installed (updates, AV software, etc.)

The endpoint will be deemed **Compliant** or **Non-compliant**

When the assessment is done, **ISE will send a CoA to the NAD** to re-authenticate and re-authorize the user

After CoA, we should then match a **NEW rule in the Authorization Policy** depending on whether the endpoint is Compliant or Non-compliant

Administration > System > Settings > Posture

General Settings:
You can set the **Remediation Timer** (how much time the user will have to fix the issues – *4 minutes by default*), and to set the **Default Posture Status** (Compliant by default). You can also automatically close the window after authentication via an option here.

Reassessments:
You can enable periodic reassessments

Acceptable Use Policy:
You can define an AUP for non-guest users (guest users have their own portal; this is just for authenticated users)

Policy > Policy Elements > Conditions

Click **Posture**, then **File Condition**. Here you can **Add** a condition to specify a particular file you can look for during posture assessment (can check for file existence or NON-existence).

Policy > Policy Elements > Results

Click **Posture**, then **Remediation Actions**, then **Requirements**. We can add a requirement called "*Posture_File_Lookup*" (or something similar). Under **User Defined Conditions**, choose the **File Conditions** created above. We can say that when the condition is NOT met, perform some action (e.g. send a message to the user).

You will also find numerous pre-defined Remediation Actions for AS, AV, File, Launch Program, Link, WSUS, Windows Update

Now we need to apply the policy via **Policy > Posture**

Add the rule name, then define the options and select the **Requirements** you defined in the previous step

Under **Policy > Policy Elements > Results > Client Provisioning > Resources**, select the **NAC Agent** and **Compliance Module** necessary to enforce your policy. You can also create an **ISE Posture Agent Profile** in this section to override some of the defaults.

Next go to **Policy > Client Provisioning** to define the NAC Agent, Posture Agent Profile (if you created one), Compliance Module, and other settings.

Go to **Policy > Policy Elements > Results > Authorization > Downloadable ACLs** to define a DACL used for Unknown posture status endpoints (you must permit DHCP, DNS, ISE, and the **SWISS protocol** (TCP/80, TCP/UDP 8905, TCP/8909, TCP/8443)

Go to **Policy > Policy Elements > Results > Authorization > Authorization Profiles** to define a profile, then choose the DACL we created above. Then specify the **Web Redirection** option for **Client Provisioning (Posture)**, along with the redirection ACL on the NAD.

FINALLY, go to **Policy > Authorization** to tie everything together! We will add a rule for Unknown, Compliant, and Non-compliant posture status.

BYOD:

One common way to allow employees to self-register their personal devices at work is via the **My Devices** portal.

My Devices Portal URL:
https://ise_ip:8443/mydevices

The **Device ID** asked for via the portal is the **MAC address** of the wired or wireless interface you are using to access the network. The **Description** field is free-form text that the user can use to describe the device they are registering – “Jane Doe’s iPhone”, etc.

After authentication, the user can also see their previously registered devices and choose the option to **Edit**, **Delete**, or report the device as **Lost**.

The end result of a user adding a device in My Devices is the device being added to the Endpoint database (Administration > Identity Management > Identities > Endpoints. ISE will automatically add the device to the **RegisteredDevices** group. The **BYODRegistration** flag will also be set to **Yes**.

You could match either of the **attributes** and build a new **Authorization Policy** rule to restrict access accordingly.

If the device is marked as **Lost**, it will match the default **Blacklist Authorization Policy rule** and access will be denied.

The second option is via an **Onboarding Process**. The device has to use a **browser** as a requirement (no game consoles, printers, etc.).

Wireless BYOD can be applied with **Single or Dual SSID**

Single SSID:
Employees connect to the same SSID for corporate and personal devices

Dual SSID:
One SSID for BYOD devices, one for corporate devices

To begin the configuration for self-service, go to:
Administration > Web Portal Management > Settings
Click **Guest**, then **Multi-Portal Configurations**, then **DefaultGuestPortal**
Click the **Operations** tab, then check the box for "***Enable Self-Provisioning Flow***"

Next, make sure Client Provisioning is enabled:
Administration > System > Settings > Client Provisioning
Enable Provisioning: Enable
If this is not enabled, we won't be able to push the supplicants to the devices

Next, we need to configure a **Native Supplicant Profile**:
Policy > Policy Elements > Results > Client Provisioning > Resources
Click **Add**, then **Native Supplicant Profile**
This will define the settings for a provisioned supplicant (Name, Description, Wired/Wireless, SSID, Security, Allowed Protocol, etc.). Supported OS's are Windows, OS X, iOS, Android (NOT Linux).

Next, we need to define the Wizard that will be used to configure the supplicant settings. In the same section, click **Add** again, then **Agent resources from Cisco site**. Check the boxes for the proper wizards for Windows / OS X (N/A for iOS / Android).

Next, we need to create Client Provisioning Policy(s):
Policy > Client Provisioning
Create the rules specifying the Wizard, and then the OS as the Condition

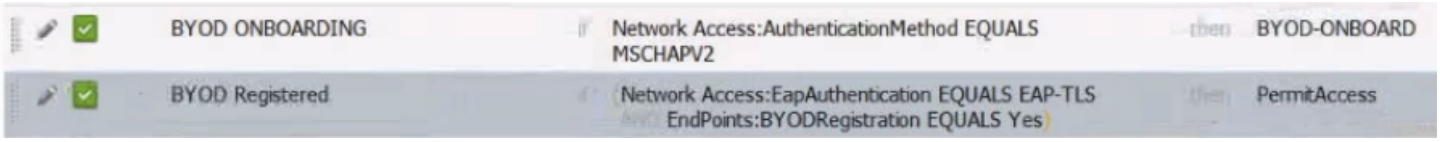
Next, we need to create Authorization Profiles:
Policy > Policy Elements > Results > Authorization > Authorization Profiles
Name it "*BYOD Onboarding*", or something similar, then specify the **Web Redirection** option for **Native Supplicant Provisioning**, along with the redirection ACL on the **WLC**.

Then, add a **second profile for BYOD devices that HAVE been successfully registered**. Specify any applicable DACL to define what the BYOD user can access once he/she has successfully completed the onboarding process.

FINALLY, go to **Policy > Authorization** to tie everything together!

Create one rule for BYOD onboarding, with a condition of the SSID used for onboarding (or if you use a Single SSID, a condition that matches non-corporate assets – something to identify a personal device). Tie this back to the onboarding Authorization Profile we created above.

Create a second rule used for BYOD-registered devices (you can use the **BYODRegistration** flag as a match). Tie this back to the registered BYOD Authorization Profile we created above.



For client-based certificates (EAP-TLS), you may need to go to **Administration > System > Certificates > SCEP RA Profiles** and specify the URL of your certificate enrollment server.

Security Group Access (SGA):

Consider a conventional access list where we had 2 different sources accessing 3 different destinations with 2 services at each destination. That would normally require 12 ACEs within an ACL.

However, with DACLs, we can specify the source as **any** because it will be dynamically substituted with the IP address of the user to which the DACL is applied. That would cut the number down to only 6 ACEs within the DACL.

The third option (instead of ACLs and DACLs) uses **Security Group Access** control. This uses a completely different approach.

There are 3 elements in this solution:

Security Group Tag (SGT):
A **16-bit value returned by ISE** upon successful login, on **INGRESS**

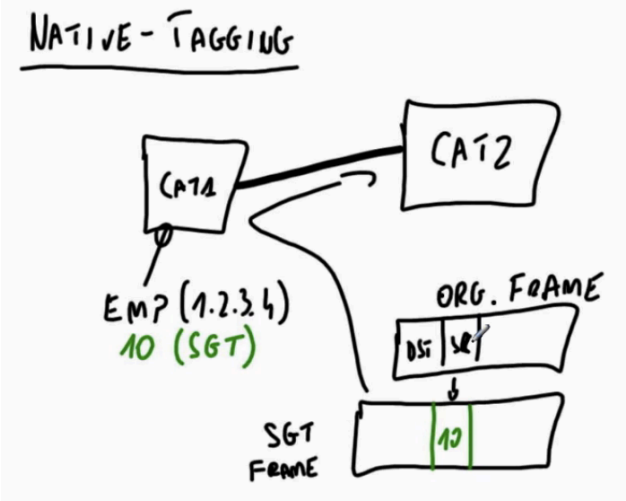
Security eXchange Protocol (SXP):
Used by non-native-tagging switches. Uses **TCP/64999**.

Security Group ACL (SGACL):
Matrix-like ACL downloaded from ISE to enforce policy on **EGRESS**

We could have, for example, a **tag** for Sales, Finance, Engineering, etc. Then we need to **propagate** information about those tags to other devices in the infrastructure. We can use this to **enforce the policy at nearly any point in the network** because all devices will know about the tags.

This normally **happens natively if your switches support SGA** with no special configuration needed. The switch will automatically include information about the SGT in the frame.

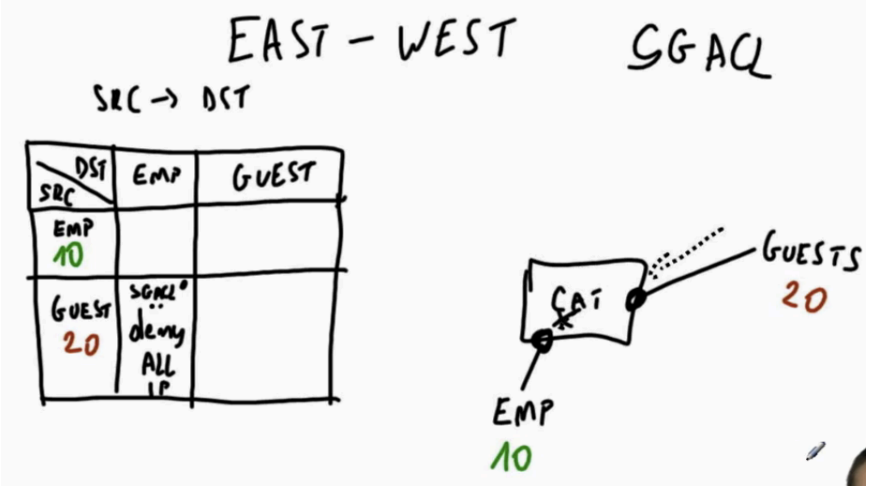
In the example below, Cat1 has a client assigned an SGT. When data is sent over the trunk to Cat2, the frame will include information about the SGT so that Cat2 knows about it.



if the **switches do NOT support SGA**, we use **SXP**. The non-native switches can learn about the tags using this protocol. The goal is to inform the switch of the **IP to SGT mapping**.

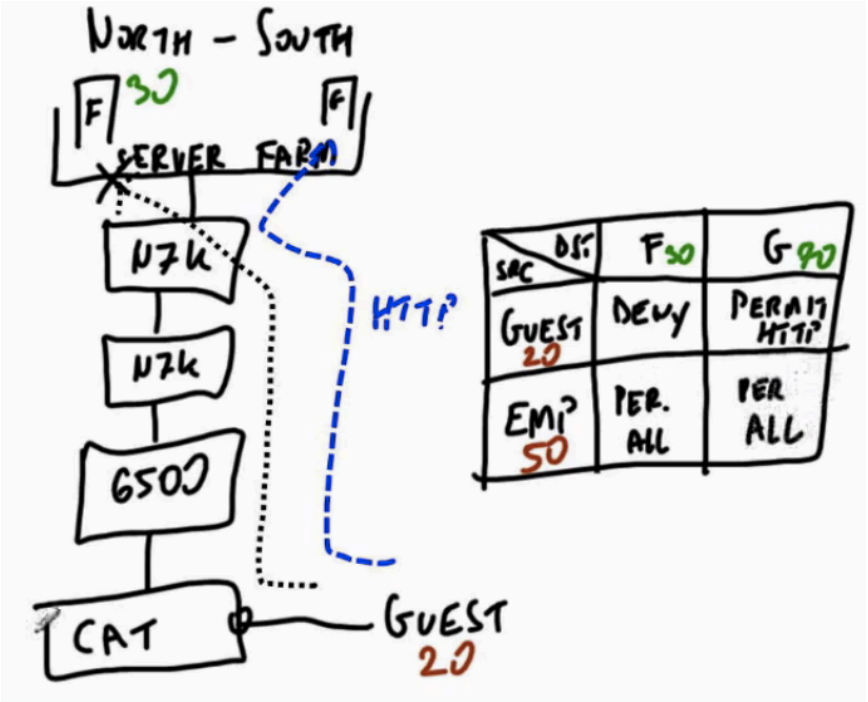
The reason why SGTs are important is because that is what is used to apply policy via SGACLs.

In the example below, Guests and Employees are in the same VLAN / IP subnet. They would normally have direct communication with one another. However, using SGACLs we can easily filter the traffic. This is called an **East-West SGACL**.



The second type of SGACL is called a **North-South SGACL**.

This could be used to protect devices in the data center. In the example below, F is Finance and G is Guest. Guests are assigned a tag of 20. Employees are assigned 50. The Finance server is assigned 30. The Guest server is assigned 70.



SGT assignment methods :

1. Dynamically - as a result of ISE Authorization
2. Configured manually on a switchport

```
cts manual
 policy static sgt tag_value
```

3. SGT-IP bindings are configured manually on ISE
   - You then need to download this information to your NADs