How to set up Cisco Secure Access Solutions with OPSWAT MetaAccess

About This Guide:

MetaAccess is a security platform that prevents risky devices from accessing local networks and cloud applications. Using OPSWAT's industry-leading vulnerabilities assessment, endpoint security and advanced threat prevention technologies, MetaAccess performs extensive security and compliance checks as well as remediation before allowing devices to access corporate data. More information about MetaAccess can be found at https://www.opswat.com/products/metaaccess.

MetaAccess can be leveraged by the Cisco ASA policies to provide enhanced compliance checking capabilities. Once you have deployed the MetaAccess agent to your devices and configured your compliance policy through the MetaAccess console, the MetaAccess agent will store the device's compliance status in the Windows Registry or Mac OS p-list. The Cisco ASA firewall can access and use this information through a *Process Scan* within the *Host Scan* configuration to determine if a device should be granted network access. The steps found within this document assume that this configuration is occurring with the ADSM console.

More information on the benefits of integrating MetaAccess with Cisco ASA can be found at https://onlinehelp.opswat.com/metaaccess/How_do_I_integrate_MetaAccess_to_my_Cisco_ASA_and_ISE_solution_.html.

Host Scan Configuration

A Cisco ASA firewall can be configured to utilize MetaAccess for advanced compliance checks for remote users via the Remote Access VPN. These checks will help enforce that endpoint devices are meeting all compliance requirements established by the organization.

The policies can be easily configured via the MetaAccess console, and ensure that the security and compliance requirements of an organization are met on a continuous basis. The Process Scan first ensures that the MetaAccess agent is actively running on the device; then the additional policy compliance state can be validated via the Dynamic Access Policies.

Process Scan

The initial configuration will focus on the Process Scan functionality. This ensures that the MetaAccess agent is running on the endpoint.

Step 1:

Validate that the MetaAccess agent is running.

To check the persistent, installed MetaAccess agent, the active process will be:

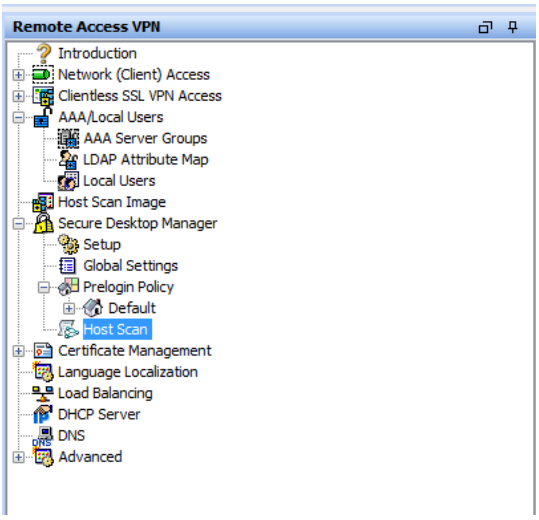
- For Windows: GearsAgentService.exe
- For Mac: opswat-gears-od

To check the on demand, portable MetaAccess agent, the active process will be:

- For Windows: opswat-gears-od.exe
- For Mac: opswat-gears-od

To configure this validation step and save for use in a Default Access Policy, go to *Remote Access VPN* within the ADSM console; and select *Host Scan* under *Secure Desktop Manager*.

Select *Add*.

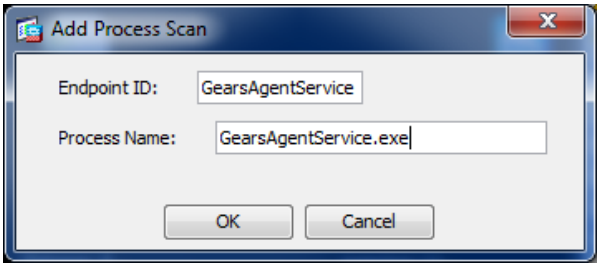


Step 2:

Select *Process Scan*, and name the *Endpoint ID* something that you will affiliate with this check (i.e. MetaAccess Service).

Enter the process name in the *Process Name* box:

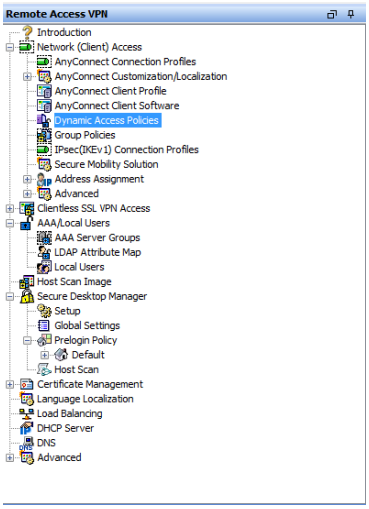
- For the persistent, installed MetaAccess agent, use GearsAgentService.exe
- For the on demand, portable MetaAccess agent, use opswat-gears-od.exe



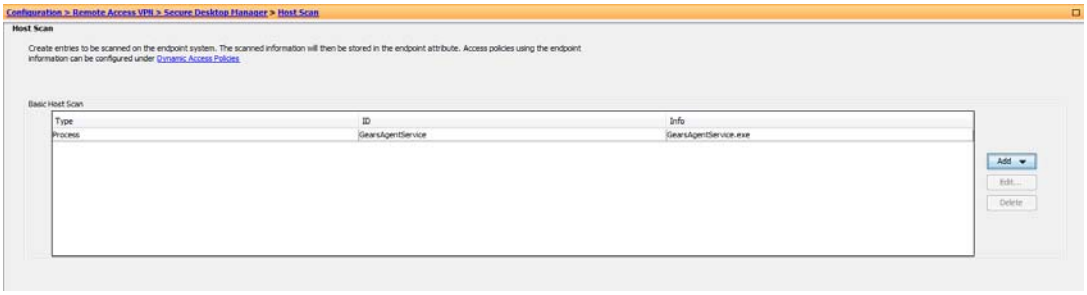
Apply these changes to your running configuration prior to continuing to the next step.

Step 3:

Navigate to *Network (Client) Access* and select *Dynamic Access Policies*.



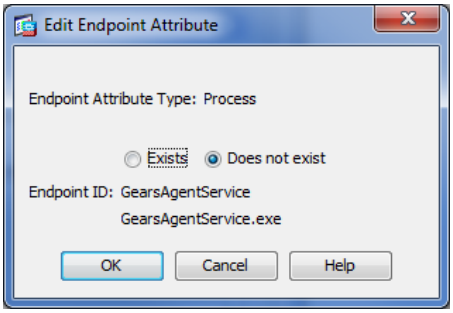
Select *Add*; then *Policy*.



Step 4:

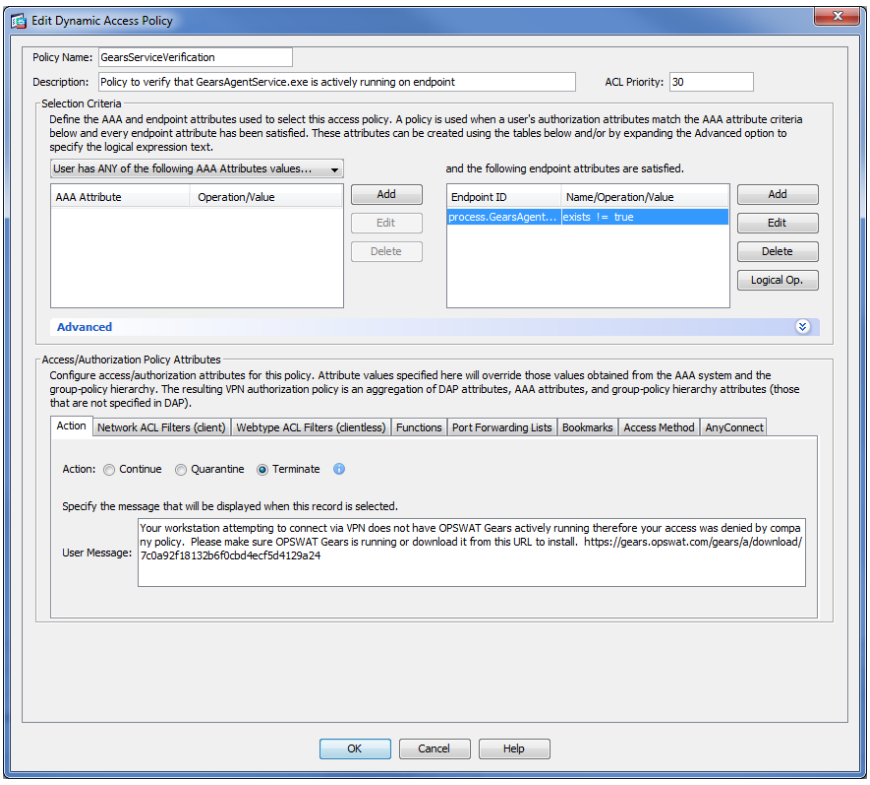
Give your new Policy a meaningful name (i.e. MetaAccessServiceVerification), and describe as appropriate. Provide an ACL priority as appropriate to your existing configuration.

Select Add in the "*and the following endpoint attributes are satisfied*". For the "*Endpoint Attribute Type*", select *Process;* then select the previously created "*Endpoint ID*" of "GEARSAgentService".



Select "*Terminate* " for the "*Action* ".

Please note, this *Dynamic Access Policy* will not allow a tunnel to be created if the MetaAccess Service is NOT running. If the process is running, you will continue to the next *Dynamic Access Policy* .



You have now configured a posture assessment and Dynamic Access Policy that will verify that the MetaAccess agent is running, prior to allowing the remote device into a network.

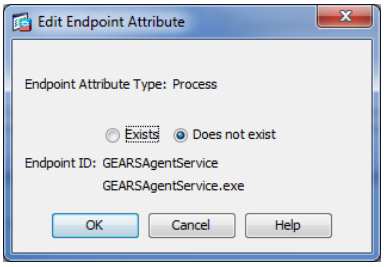
Registry Scan

The next configuration step is to establish the Registry Scan. This ensures that the endpoint meets the predefined compliance requirements prior to allowing access to the network.

The following steps will outline the how to establish the registry check for both 32-bit and 64-bit Windows devices.

Step 1:

Edit the *Endpoint* Attribute previously created – "MetaAccessService". Establish the *Registry Scan* for the 64-bit system by first creating the *Endpoint Name* for the check. This name should be unique to designate the difference between the two checks (32-bit versus 64-bit). Now add requirements for the following *Registry Scan* details.



Confirm the Registration Key on the Client matches the Account.

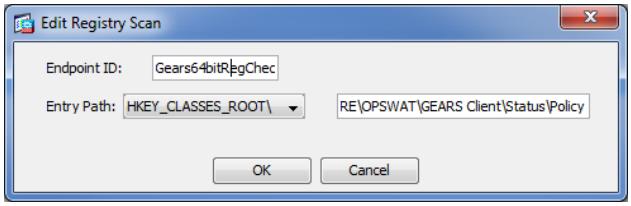
1. For the persistent, installed MetaAccess agent:
   - Registry root key – HKEY_LOCAL_MACHINE"
   - Registry subkey – \SOFTWARE\Wow6432Node\OPSWAT\Gears Client\Config"
   - Name – RegistrationKey
   - Type – REG_SZ
   - Value should match the account Registration Key"
2. For the on demand, portable MetaAccess agent:
   - Registry root key – HKEY_CURRENT_USER
   - Registry subkey – \SOFTWARE\OPSWAT\Gears OnDemand\Config
   - Name – RegistrationKey
   - Type – REG_SZ
   - Value should match the account Registration Key

Check the Compliance state on the endpoint.

1. For the persistent, installed MetaAccess agent:
   - Root key – HKEY Local Machine
   - Subkey – \SOFTWARE\Wow6432Node\OPSWAT\Gears Client\Status
   - Name – Policy
   - Type – DWORD
   - Value – 0x0000000 [1]
2. For the on demand, portable MetaAccess agent:
   - Root key – HKEY Current User
   - Subkey – \SOFTWARE\OPSWAT\Gears OnDemand\Config
   - Name – Policy
   - Type – DWORD
   - Value – 0x0000000 [1]

Policy Key Values:

a. 0 = NOT in compliance with a defined policy on your account, check the MetaAccess console for details on the device
b. 1 = in compliance with a defined policy on your account, check the MetaAccess console to view the defined policy



Select *Ok* to save the changes to enable the check for a 64-bit registry.

Step 2:

Repeat Step 1 to create an additional *Endpoint Attribute* for the 32-bit Registry, but using the following *Registry Scan* details.

Confirm the Registration Key on the Client matches the Account.

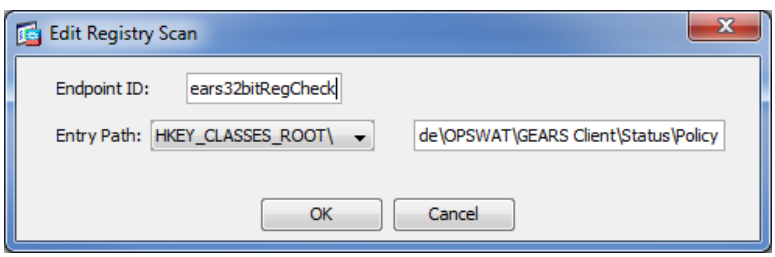
1. For the persistent, installed MetaAccess agent:
   - Registry root key – HKEY_LOCAL_MACHINE"
   - Registry subkey – \SOFTWARE\OPSWAT\Gears Client\Config"
   - Name – RegistrationKey
   - Type – REG_SZ
   - Value should match the account Registration Key"
2. For the on demand, portable MetaAccess agent:
   - Registry root key – HKEY_CURRENT_USER
   - Registry subkey – \SOFTWARE\OPSWAT\Gears OnDemand\Config
   - Name – RegistrationKey
   - Type – REG_SZ
   - Value should match the account Registration Key

Check the Compliance state on the endpoint.

1. For the persistent, installed MetaAccess agent:
   - Root key – HKEY Local Machine
   - Subkey – \SOFTWARE\OPSWAT\Gears Client\Status
   - Name – Policy
   - Type – DWORD
   - Value – 0x0000000 [1]
2. For the on demand, portable MetaAccess agent:
   - Root key – HKEY Current User
   - Subkey – \SOFTWARE\OPSWAT\Gears OnDemand\Config
   - Name – Policy
   - Type – DWORD
   - Value – 0x0000000 [1]

Policy Key Values:

a. 0 = NOT in compliance with a defined policy on your account, check the MetaAccess console for details on the device.
b. 1 = in compliance with a defined policy on your account, check the MetaAccess console to view the defined policy.

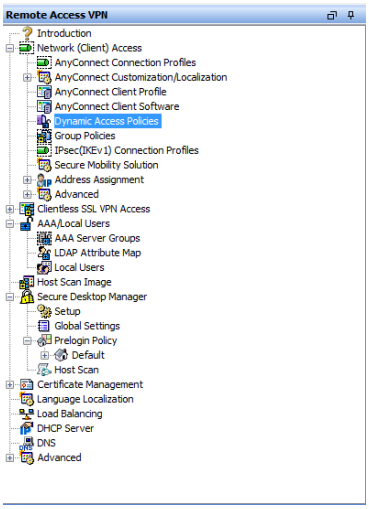


Select *Ok* to save the changes to enable the check for a 32-bit registry.

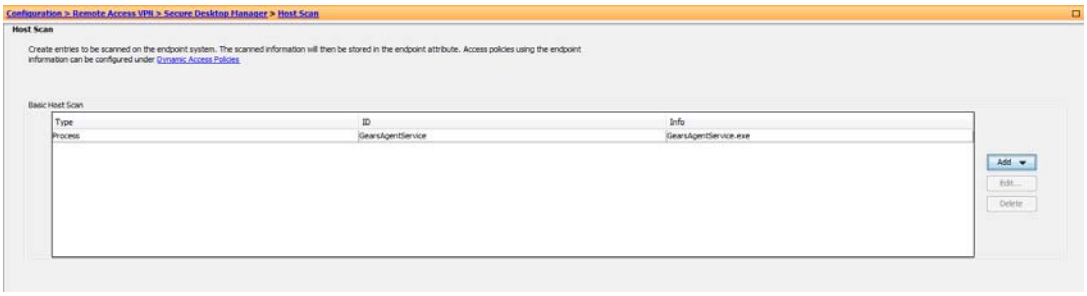
Step 3:

For Mac devices, the client provides a file with the Registration Key and Policy value. To configure for the Mac, first you need to create a new *Host Scan*.

Navigate to *Secure Desktop Manager* and *Host Scan*.



Under *Host Scan*, select *Add*; then *File*.



Step 4:

Add a new file.

- For the persistent, installed MetaAccess agent, use the file: Applications/OPSWAT GEARS Client/Policies/GEARS_<MetaAccess license key>_<0 or 1>.txt.
- For the on demand, portable MetaAccess agent, use the file*:/Users/<username>/Documents/OPSWAT/GEARS OnDemand/ GEARS_<MetaAccess_license_key>_<0 or 1>*

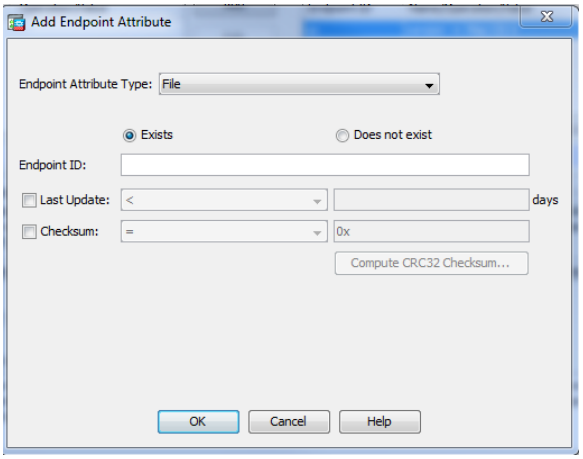
The *MetaAccess license key* will be where you add your *Account Registration Key* , and the "1" represents the Policy Value of a device that passes the policy defined in the MetaAccess console.

This file includes a combination of 2 values, Policy and LicenseKey, to ensure that the client installed is assigned to the Account that manages the defined Polices.

Step 5:

Add the *Endpoint Attribute* based on the file you previously created – *Endpoint Attribute Type* – File. The value that will be shown, is the full path added when the File was created.

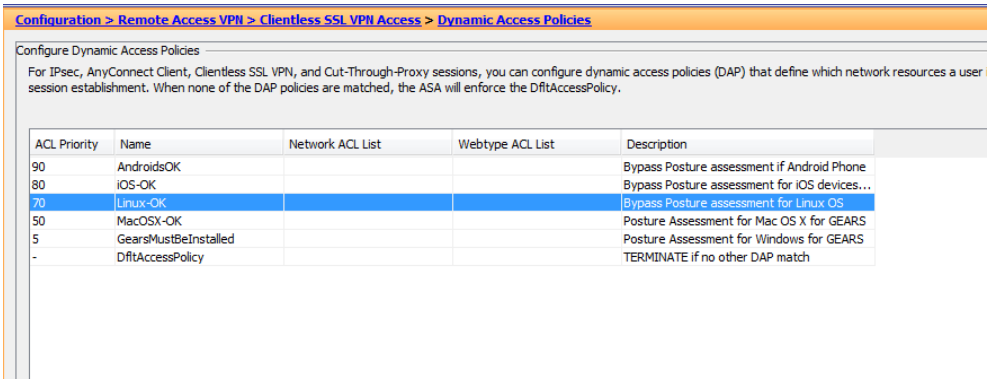
- For the persistent, installed MetaAccess agent, use the file: Applications/OPSWAT GEARS Client/Policies/GEARS_<MetaAccess license key>_<0 or 1>.txt.
- For the on demand, portable MetaAccess agent, use the file*:/Users/<username>/Documents/OPSWAT/GEARS OnDemand/ GEARS_<MetaAccess_license_key>_<0 or 1>*



Select OK.

Step 6:

Complete setup of any other requirements you wish to include in the Host. Once completed, go to *Dynamic Access Policies* to determine the priority of the *Policies*.



For more information, or if you have any questions about the steps above, please log into the OPSWAT Portal at https://my.opswat.com and submit a ticket to request assistance from our support team.