



Deployment Planning

Overview

This section provides information on deploying the product as part of pilot program and scaling the product to 100,000 agent deployments.

This section contains the following topics:

- [Piloting the Product, page 2-2](#)
- [Running a Pilot Program, page 2-2](#)
- [Scalable Deployments, page 2-3](#)
- [Hardware Sizing, page 2-3](#)
- [Software Considerations, page 2-5](#)
- [Configuration Recommendations for Scalability, page 2-5](#)
- [Policy Migration, page 2-6](#)
- [Configuration Item Mapping, page 2-6](#)
- [Policy Tuning and Troubleshooting, page 2-7](#)
- [Overall Guidelines, page 2-7](#)
- [Using Test Mode, page 2-10](#)
- [Disabling Specific Rules, page 2-11](#)
- [Caching and Resetting Query Responses, page 2-11](#)
- [Setting Up Exception Rules, page 2-12](#)

Piloting the Product

Before deploying Cisco Security Agents (CSA) on a large scale, it is critical that you run a manageable and modest initial pilot of the product. Even in a CSA upgrade situation, a pilot program is required. Due to the unique configuration of every individual enterprise, the pre-configured policies that ship with CSA will not fit every site perfectly. A certain amount of policy tuning is always necessary. This tuning is best done on a small sample of systems that are representative of the whole.

Once the pilot is operating satisfactorily, with CSA protecting systems using properly tuned policies, you can turn your pilot into a larger deployment.

The following sections provide a guideline for conducting a pilot of CSA and deploying the product on a large scale.

Running a Pilot Program

Your pilot program should proceed in the following manner:

- *How large should a pilot program be?* Select a logical, manageable, sample of systems on which agents will be installed. A good rule of thumb is to make your pilot approximately one /one-hundredth the size of what the entire deployment will be.

Details:

- If your entire deployment will be very small, be sure to pilot at least 15-20 systems.
- If your entire deployment will be very large, roll out your pilot in steps. For example, do not pilot 1,000 systems initially and all at once. Start with a smaller sample and gradually expand the pilot.

The pilot should include machines that you can access readily (either yourself or through a responsive end-user). If you will eventually be installing agents on multiple, supported operating systems, your pilot should include machines running those operating systems. Again, systems in your pilot should be representative of the whole deployment to which you intend to scale.

- *How long should a pilot program run?* Basically, the deploying and tuning of policies is an iterative process. Initially, you will have a great deal of event log noise to parse. You must examine the data coming in and edit your policies accordingly.

Details:

- Although every site is different, it would not be unusual to run a pilot program for approximately 90 days. All possible application usage should take place within the pilot time frame. It is important to note that this recommended time frame allows you to exercise applications, their deployment and usage, within an entire fiscal quarter. The idea being, every application you use and every manner in which you use it will occur during this piloting period.

Scalable Deployments

The Cisco Security Agent V5.0 release offers scaling of agents to 100,000 systems. To reach this deployment number, there are recommended multi-tiered CSAMC server system hardware, CPU, and memory requirements. Please refer to the following section.

Hardware Sizing

This section provides three server configuration examples and three hardware configuration examples. The server and hardware combinations will be charted in three tables providing information on how many agents can be deployed using each server and hardware configuration combination. This should give you an idea of how to configure CSA to scale up to a 100,000 agent deployment.

For the purpose of this guide, we will use three server configuration examples.

Server Configurations:

1. Single server
2. Two servers: one server for polling and configuration, one database server
3. Three servers: one server for polling, one server for configuration, one database server

We will use the following hardware configurations.

Hardware Configurations:

1. Single processor Pentium 4 (3Ghz+) with 2 GB RAM
2. Dual processor Xeon (2.5 Ghz+) with 4 GB RAM
3. Quad processor Xeon (2.5 Ghz+) with 8 GB RAM
4. Eight-Way Xeon (2.5 Ghz+) with 8 GB RAM

The following tables approximate the number of agents you could deploy with each server configuration installed on one of four hardware configurations provided.

Table 2-1 Server Configuration 1: Single Server

Hardware Configuration	Number of Agents
Hardware Configuration 1	2,500
Hardware Configuration 2	5,000
Hardware Configuration 3	10,000
Hardware Configuration 4	20,000

Table 2-2 Server Configuration 2: Two Servers

Hardware Configuration	Number of Agents
Hardware Configuration 1	7,500
Hardware Configuration 2	15,000
Hardware Configuration 3	30,000
Hardware Configuration 4	75,000

Table 2-3 Server Configuration 3: Three Servers

Hardware Configuration	Number of Agents
Hardware Configuration 1	10,000
Hardware Configuration 2	20,000
Hardware Configuration 3	50,000
Hardware Configuration 4	100,000

Software Considerations

- CSA MC is only supported on Windows 2000 Server and Windows 2000 Advanced Server operating systems at this time. Only Hardware Configurations 1 and 2 (referenced in previous tables) support Windows 2000 Server. Hardware Configuration 3 with 8GB RAM requires Windows 2000 Advanced Server to take advantage of the increased memory. Refer to the Microsoft web site product information section for details.
- Supported server operating systems should have a minimum patch level of Service Pack 4 with Rollup 1.
- To support any deployment over 500 agents, you should use Microsoft SQL Server 2000 in lieu of MSDE. Only Hardware Configuration 1 supports Microsoft SQL Server 2000 Workgroup or Standard editions with their 2GB RAM limitation.
- The Microsoft SQL Server 2000 should be patched to Service Pack 3a. Later service packs for SQL Server are not qualified.



Note Your memory consumption needs should dictate your CSA MC operating system choice, i.e. Windows 2000 Server or Windows 2000 Advanced Server.

Configuration Recommendations for Scalability

If you intend to scale to a deployment of approximately 100,000 agents, there are some configuration recommendations you should consider.

Set Polling Interval

With 100,000 agents deployed across your enterprise, you want to ensure that no more than 20 agents are communicating with the MC approximately every second or so. Therefore, with a deployment of this size, it is recommended that you set the polling interval to no less than 1 hour. You can have some systems polling in every hour and others polling in later than that. But on average, a 1 hour or higher polling interval is appropriate. Be sure to have the polling hint functionality enabled, as well.

Use Content Engines

For large deployments, it is highly recommended that you use content engines with transparent web caching. It makes sense to direct groups of agents to different content engines in large deployment scenarios. Content engines reduce the load on the MC by caching rule downloads and software updates.

Policy Migration

If you are upgrading to CSA MC V5.0 from a previous version, you may want to continue to use policies that you've created and deployed successfully with CSA MC V4.x. Because some configuration items, such as action options and application classes, have changed in V5.0, you cannot do a straight import of your existing configurations into V5.0 and expect them to behave as they did previously.

The next section provides information on configuration item mapping from previous CSA MC versions to this latest release. Because feature implementation has changed in V5.0, you should plan to pilot even those imported configuration items that you've successfully run with a previous version. CSA MC V5.0 contains significant changes which will require some administrator oversight that is best performed in the process of a pilot program.

Configuration Item Mapping

You should refer to *Using Management Center for Cisco Security Agents 5.0* for details on the hierarchical changes that have been made to the CSA MC configuration process. More specifically, the following features that appeared in V4.5 now appear in V5.0 as follows:

- The Set action functionality is now used to perform some of the Add/Remove built-in application process tagging that was available in previous releases. The following built-in applications that were available in previous releases have been removed for V5.0 and although the functionality is the same, they are now implemented using the Set function: Authorized rootkit, Unauthorized rootkit, Processes communicating with Untrusted Hosts, Processes requiring Security Level <High, Medium, Low>.
- Although the functionality and general precedence of rule action types has not changed, the names have been altered. The “High priority” actions are now named “Priority”.

**Note**

This is not a listing of new features in CSA MC V5.0. This is simply a list of items that were previously in the product and have been changed in a significant manner in this release. This change will affect any existing configurations you import from a previous version into CSA MC V5.0. Therefore, you should double-check and possibly tweak affected, imported items to correctly adapt them to V5.0.

Policy Tuning and Troubleshooting

Once you have started your CSA pilot, you need to tune the policies to suit your needs and troubleshoot any problems that occur.

Overall Guidelines

This section presents some overall guidelines for tuning and troubleshooting your CSA pilot. Please read through this section carefully and consider the specific needs and requirements of your pilot before moving on to actually using the techniques. Here are the most important guidelines to follow when tuning and troubleshooting policies:

- *Never directly modify one of the supplied groups, policies, or rule modules.* If you need to change a group, policy, or rule module, make sure you *clone and rename* it first so you preserve it for use later. Modifying the supplied groups, policies, and rule modules directly makes it difficult to back out of any inadvertent mistakes.
- Use the supplied groups and if necessary define additional groups for *each distinct desktop and server type* in your network. In your pilot, you should have some participants that are using each desktop and server type so you can tune and troubleshoot all policies before deployment.

Group membership is cumulative, which can be useful in tuning and troubleshooting. For example, at the beginning of a pilot, participating hosts that are Windows desktops would be attached to the **All Windows and Desktops - All Types** groups on the **Systems -> Groups** menu. Once you have tuned the basic desktop policies, you might attach some of those hosts to the **Desktops - Remote or mobile** group. Once you are satisfied with the

performance of the remote/mobile policies, you could define a new group for a specific department's applications, attach hosts to the new group, and pilot those policies.

- Start piloting all groups in *test mode* and examine the event log (**Events -> Event Log** menu) for possible tuning and troubleshooting needs before moving to enforcement mode (also known as live mode). With the current release, you can place *all policies for a group* in test mode or a *single rule module* in test mode. Therefore, as you tune and troubleshoot, you can incrementally move rule modules to enforcement mode if need be. Keep in mind when using test mode that the area under test is completely vulnerable from a security standpoint.
- Policy tuning and troubleshooting is an *iterative* process. Focus on a single policy for improvement at a time and then verify that the tuning and troubleshooting techniques did what you expected before deploying the improved policy.
- *Prioritize* the security features you want to implement with CSA policies. You can also prioritize applications and groups. By having clear priorities and working through a single policy improvement at a time, you can manage the complexity of deploying large policy sets in large networks. For example, based on priorities, you can keep a specific rule module in test mode while the rest of the rule modules in the policy are in live mode.
- Large policy sets can generate enormous numbers of log messages, so you need to use the tools provided that help *filter out* extraneous information and *isolate* the specific policy to be improved or behavior to be studied. For example, you can log only the events that result in Deny actions or create an exception rule that stops logging a specific event to reduce the overall number of log messages. In addition, host diagnostics can be used to filter rules based on the user state (that is, the user and group) the host is in, such as only logging the behavior of the rules used by members of the Administrator group. Monitor policies can be used in clever ways to focus in on specific behavior without interrupting applications and services.
- Set up *separate agent kits* to support the different features of your pilot. For example, you might have some desktop kits that have all policies in test mode, some desktop kits with a basic set of well-tested policies in live mode plus one experimental policy in test mode, and so forth. Labelling these kits clearly will help your pilot participants download the right set of policies you want to test and give you clear feedback on areas needing improvement.

There are two general approaches to policy creation, and the approach you choose affects how you tune and troubleshoot the policies:

- Using the *supplied* Desktop and Server group policies plus a few application-specific policies. In this scenario, you attach each participating host to the following groups:
 - <All <platform>>
 - **Desktops - All types** or **Servers - All types**
 - A task-specific group, such as **Servers - Apache Web Servers** or **Servers - SQL Server 2000**

Then, you attach each group to the following policies:

- A **Virus Scanner** policy. CSA supplies policies for Norton, McAfee, and Trend antivirus software. If you are using a different antivirus product, you might need to use the generic Virus Scanner policy, or clone it and make modifications to suit your virus scanner application.
- An **Installation Applications** policy. CSA supplies installation software policies for Windows, Linux, and Solaris.

**Note**

If you do not attach antivirus and installation policies to each participating group of hosts, the CSA event logs will contain a large number of false positives, making it difficult to manage the pilot.

After attaching the Desktop and Server groups, Virus Scanner policy, and Installation Application policy, you are ready to create agent kits, start the pilot, examine the event log, and stage the next policy additions. For example, if you have a prioritized list of applications to protect, start with the first on the list, use the **Analysis -> Application Behavior Investigation** tool to understand the behavior of the application, craft a policy, place it in test mode on the pilot machines, and examine the event log. Use the techniques in the rest of this section to tune/troubleshoot that application's policy, re-examine the event log, and if you are satisfied with the result, place the application's policy in live mode on the pilot machines. You repeat these steps with each application on your prioritized list.

- Creating a completely *custom* set of policies. In this scenario, you have a team of network security experts who have assembled a detailed list of security features and studied the many supplied rule modules. The experts use the **Analysis -> Application Behavior Investigation** tool to thoroughly study

the applications for which they will write rules. Then, the experts will craft custom policies by selecting the desired rule modules and rules. With this custom approach, consider conducting a small pilot of a few systems in a test lab and then expanding to a larger and more thorough pilot.

Using Test Mode

CSA policies can execute in *live mode*, where they enforce rules by denying or allowing events, or *test mode*, where they indicate in the event log what the action would have been to the given event. All entries in the event log for rules in test mode begin with the label `TESTMODE:` to make it easy to scan for events relating to rules under test. In general, you start a pilot in test mode and gradually change over to live mode as you examine the performance of each policy. You can use test mode in two different ways:

- Place *all policies for a group* in test mode.

From the **Systems->Groups** menu, you use the supplied **Systems - test mode** group, which is available for Windows, Linux, and Solaris. You attach hosts (both desktops and servers) to each appropriate test mode group. You can make one or more agent kits available for download with the test mode groups. Be sure to include “test mode” in the name of the agent kit.

When the “test mode” phase of the pilot is completed, you can unattach hosts from the test mode groups to place the hosts in live mode.

- Place *a specific rule module* in test mode.

If one of the rule modules within a policy is not behaving as expected, you can place it in test mode while still keeping the remaining rule modules in live mode. To do this, select the **Test Mode** checkbox on any **Configuration -> Rule Modules -> <platform> Rule Modules -> <module name>** page.



Note

When running your pilot, explain to participants the difference between test mode and live mode, clearly label whether agent kits are for test mode or live mode, and tell participants which kits to download and use during various phases of the pilot.

Test mode is *not* intended to be used indefinitely because the area under test is completely vulnerable from a security standpoint. Groups and rule modules in test mode should move to live mode in a timely fashion. Once the pilot is over, you need to carefully control which hosts if any are in test mode. You can remove the test mode kits to ensure they do not get downloaded during deployment and

periodically monitor the **Systems - test mode** group to ensure that all pilot participants have migrated to live mode agent kits. You want to avoid the situation where a security hole exists after deployment because some groups or rule modules were inadvertently left in test mode.

Disabling Specific Rules

When you examine the event log with the **Events -> Event Log** menu, the description of each event references the *rule number*. If you find a consistent pattern of false positives with the same specific rule number, you can disable that rule if desired. There are two different approaches to disabling rules:

- You can disable the rule *temporarily*. At a later time, you can go back and modify the rule, set up a query with a cached response, or set up an exception rule.
- You can disable the rule *permanently* if the rule protects a resource that you don't need protected as part of your security policy.

The easiest way to disable a rule is by clicking on the rule number at the bottom of the event description in the event log. On the rule page, you click on the Enabled checkbox to uncheck it and disable the rule. Once you generate the rules, this rule will be disabled.

Caching and Resetting Query Responses

Rules can be configured with enforcement actions of allow, deny, terminate, or query the user. In some cases, there are rules that already query the user but do so repeatedly instead of caching the user's response to make it persistent. In other cases, there are rules that are generating a mix of false positives and valid enforcements in the event log and need to be modified so they query the user and cache the user's response for the false positives.

You set up a query and cache the answer with *different* MC menus:

- To set up a query, you display the rule you wish to modify by clicking on the rule number in the event log. You then select **Query User** from the action popup menu.

- To cache the response for a query, select the **Configuration -> Variables -> Query Settings** menu option, and then select the desired query from the page. Then, click on the **Enable “don’t ask again” option** checkbox if it is not already checked. When users receive the query and indicate they don’t want to be asked this query again, their answer is cached.

**Note**

One trade-off of setting up a cached query response is that users can answer the query inappropriately and then the inappropriate response becomes persistent. After setting up a cached query response, review the event log to make sure users are responding appropriately to the query. If some users give inappropriate responses, you can reset their agents and then give the users more information about responding to the query.

If a user has responded to a query inappropriately and the response is being cached, you can reset the user’s cache by doing the following:

1. Select the **Systems -> Hosts** menu option.
2. Click on the `<hostname>`.
3. Select **User Query Responses** and click on the **Reset Cisco Security Agent** button.

Setting Up Exception Rules

In some cases, you need two or more different rules to completely specify the desired actions to a specific event. For example, you could have one rule that denies all applications from writing to the `//blizzard/webdocs` directory and another rule that allows the WebGuru application with authenticated user webmaster to write to the `//blizzard/webdocs` directory. The second rule allowing write access for WebGuru is considered *an exception rule* because it overrides a small part of the overall deny rule for the `//blizzard/webdocs/` directory. The MC manipulates the precedence of exception rules so that they are evaluated before the rules that they override.

Although you can create exception rules with the MC rule pages, the easiest way to create exception rules is using the Event Management Wizard from the event log. The wizard tailors its behavior to the event from which you launch it. You can use the wizard to create two general types of exception rules:

- Exception rules that under certain conditions allow an event that was denied

- Exception rules that stop logging similar events

To launch the wizard:

1. Select **Events -> Event Log**.
2. Click on the **Wizard** link at the bottom of the desired event's description.

The wizard asks you questions about the following:

- Whether the exception rule applies to the user/state conditions of the triggering rule or the user/state conditions of the specific event where you launched the wizard. If you want the exception to apply to all users, you typically want the user/state conditions of the triggering rule (the default). If you want to create an exception rule only for the user specified in the event, you need to explicitly select the **specific user state conditions** radio button
- Whether the description of the proposed exception rule looks correct. Keep in mind that if you need to make some small changes to the rule, such as the applications specified, you can do so later. After the wizard finishes, you can still modify the exception rule further before saving it.
- Whether you want to put this new exception rule in a separate exception rule module (the default) or modify the rule module that triggered the event. In most cases, you want to put this in a separate exception rule module so you can preserve the supplied rule modules.
- Whether you want the exception rule based on the application specified in the event or whether you want to base it on a new application class.

After you click Finish in the wizard, the MC displays the new exception rule. At this point, you should do the following:

1. Change the **Description** field to an appropriate name.
2. Examine the details in the **when** box. If necessary, you can change these details to expand or narrow the conditions for the exception.
3. Click the **Save** button.

