配置和识别安全终端排除

目录

简介

免责声明

概述

什么是排除项

- 思科维护的例外项
- 自定义排除

排除的类型

- 进程排除
  - MacOS和Linux
  - Windows 窗口版本
- 威胁排除
- 路径排除
  - 部分路径匹配（仅限Windows）
- 文件扩展名排除
- 通配符排除
  - Windows 窗口版本
- 可执行文件排除项（仅限Windows）
- IOC排除（仅限Windows）

CSIDL和KNOWFOLDERID（仅Windows）

准备用于排除调整的连接器

确定排除项

- MacOS和Linux
  - 创建进程排除
  - 创建路径、文件扩展名和通配符例外项
  - 行为保护引擎
- Windows 窗口版本

在安全终端控制台中创建排除规则

最佳实践

- 不推荐的排除项

相关信息

简介

本文档介绍例外项、如何识别例外项，以及在思科安全终端上创建例外项的最佳实践。

免责声明

本文档中的信息基于Windows、Linux和macOS操作系统。

本文档中的信息都是基于特定实验室环境中的设备编写的。本文档中使用的所有设备最初均采用原始（默认）配置。如果您的网络处于活动状态，请确保您了解所有命令的潜在影响。

概述

阅读本文档后，您必须了解：

- 什么是例外以及思科安全终端可用的不同类型例外项。
- 如何准备连接器以进行排除调整。
- 如何识别潜在的强排除项。
- 如何在思科安全终端控制台中创建新例外项。
- 创建排除项的最佳实践是什么。

什么是排除项

排除集是目录、文件扩展名、文件路径、进程、威胁名称、应用程序或不希望连接器扫描或定罪的危害表现列表。当启用终端保护（如安全终端）时，需要精心编制例外项，以确保计算机上的性能和安全性达到平衡。本文描述安全终端云、TETRA、SPP和MAP的例外情况。

每个环境都是独特的，而且控制它的实体也各不相同，从严格的策略到开放的策略。因此，除外情况必须针对每种情况量身定制。

排除项可按两种方式分类：思科维护的排除项和自定义排除项。

思科维护的例外项

思科维护的例外项是基于研究而创建的例外项，并且已经对常用操作系统、程序和其他安全软件进行了严格测试。可以通过在“排除”(Exclusions)页面的“安全终端控制台”(Secure Endpoint Console)中选择“思科维护的排除”(Cisco-Maintended Exclusions)来显示这些排除：

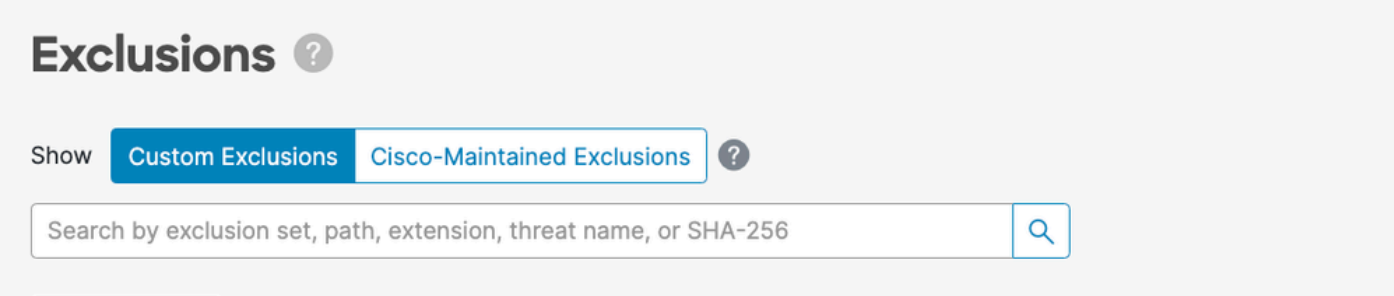


思科监控防病毒(AV)供应商发布的推荐排除列表，并更新思科维护的排除项以包括推荐的排除项。

注意：某些AV供应商不发布推荐的例外项。在这种情况下，您需要联系AV供应商请求推荐的例外项列表，然后打开支持案例以更新思科维护的例外项。

自定义排除

自定义例外项是用户为终端上的自定义使用案例创建的例外项。通过在Exclusions页面上的安全终端控制台中选择Custom Exclusions，可以显示这些例外项。



排除的类型

进程排除

进程排除允许管理员从支持的引擎中排除进程。下表列出了每个平台上支持Process exclusions的引擎：

操作系统	引擎			
	文件扫描	系统进程保护	恶意活动保护	行为保护
Windows 窗口版本	✓	✓	✓	✓
Linux	✓	✗	✗	✓
macOS	✓	✗	✗	✓

MacOS和Linux

创建进程排除时必须提供绝对路径，也可以提供可选用户。如果同时指定路径和用户，则必须满足两个条件才能排除进程。如果不指定用户，则进程排除将应用于所有用户。

注意：在macOS和Linux上，进程排除项适用于所有引擎。

进程通配符：

安全终端Linux和macOS连接器支持使用进程排除中的通配符。这允许更宽泛的覆盖范围，较少地排除某些内容，但如果对太多内容不加界定，也可能会带来危险。

警告：您只能使用通配符来包含提供所需例外项所需的最小字符数。

进程通配符用于macOS和Linux:

- 通配符用单个星号字符(*)表示。
- 可以使用通配符代替单个字符或完整目录。
- 将通配符置于路径开头将被视为无效。
- 通配符在两个定义的字符、斜杠或字母数字之间起作用。

示例：

排除	预期结果
/Library/Java/JavaVirtualMachines/*/java	排除JavaVirtualMachines的所有子文件夹中的java
/Library/Jibber/j*bber	不包括Jabber、Jabber、Jobber等进程

Windows 窗口版本

创建进程排除时，您可以提供进程可执行文件的绝对路径和/或SHA-256。如果同时指定路径和SHA-256，则必须满足这两个条件才能排除该进程。

在Windows上，还可以在路径中使用CSIDL或KNOWNFOLDERID来创建“进程排除”。

⚠ 警告：默认情况下不排除由排除进程创建的子进程。要在创建进程排除时排除其他进程，请选择应用到子进程。

限制:

- 如果进程的文件大小大于策略中设置的最大扫描文件大小，则不计算进程的SHA-256，并且排除不起作用。对大于最大扫描文件大小的文件使用基于路径的进程排除。
- Windows连接器在所有进程排除类型中限制为500个进程排除。
  - 从policy.xml中的进程排除列表顶部开始，进程排除数只满足限制。
  - 每个Windows策略都有一个针对sfc.exe的“进程”排除项，该排除项根据进程排除项限制计算：

```
3|0||CSIDL_Secure Endpoint_VERSION\sfc.exe|48|
```

✎ 注意：在Windows上，每个引擎都会应用进程排除项。如果相同排除项必须应用于多个引擎，则在这种情况下，对于每个适用的引擎都必须复制进程排除项。

进程通配符：

安全终端Windows连接器支持使用进程排除中的通配符。这允许更宽泛的覆盖范围，较少地排除某些内容，但如果对太多内容不加界定，也可能会带来危险。

⚠

警告：您只能使用通配符来包含提供所需例外项所需的最小字符数。

Windows进程通配符的使用：

- 通配符用单个星号字符(*)或双星号(**)表示
- 单星号通配符(*):
  - 可以使用通配符代替单个字符或完整目录。
  - 将通配符置于路径开头将被视为无效。
  - 通配符在两个定义的字符、斜杠或字母数字之间起作用。
  - 将通配符置于路径末尾会排除该目录中的所有进程，但不包括子目录。
- 双星号通配符(**):
  - 只能放置在路径的末尾。
  - 将通配符置于路径末尾会排除该目录中的所有进程以及子目录中的所有进程。
  - 这允许使用最小输入创建更大的排除集，但也会为可视性留下很大的安全漏洞。请谨慎使用此功能。

示例：

排除	预期结果
C:\Windows*\Tiworker.exe	排除在Windows子目录中找到的所有Tiworker.exe进程
C:\Windows\P*t.exe	不包括Pot.exe、Pat.exe、P1t.exe等
C:\Windows*keys.exe	排除Windows目录中以keys.exe结尾的所有进程
C:*	不包括C:驱动器，但不在子目录中
C:**	排除C上的每个进程：驱动器

威胁排除

通过威胁排除，您可以在触发事件时排除特定威胁名称。只有当您确定事件是误报检测的结果时，才必须使用“威胁排除”。在这种情况下，使用事件中的确切威胁名称作为您的威胁排除。请注意，如果您使用此类排除，则甚至不会检测、隔离或生成对威胁名称的真正检测。

注意：威胁排除不区分大小写。示例：W32.Zombies.NotAVirus和w32.zombies.notavirus都匹配相同的威胁名称。

警告：除非经过彻底调查确认威胁名称为误报，否则不要排除威胁。排除的威胁不再填充事件选项卡进行检查和审核。

路径排除

路径排除最常用，因为应用程序冲突通常涉及目录排除。可以使用绝对路径创建路径排除。在Windows上，您还可以使用CSIDIL或KNOWNFOLDERID创建路径排除项。

例如，要在Windows的Program Files目录中排除AV应用程序，排除路径可以是以下任何路径：

```
C:\Program Files\MyAntivirusAppDirectory
CSIDL_PROGRAM_FILES\MyAntivirusAppDirectory
FOLDERID_ProgramFiles\MyAntivirusAppDirectory
```



注意：路径排除是递归的，也排除所有子目录。

部分路径匹配（仅限Windows）

如果路径排除中未提供尾随斜杠，则Windows连接器对路径执行部分匹配。



注意：Mac和Linux不支持部分路径匹配。

例如，如果您在Windows上应用以下路径排除项：

```
C:\Program Files
C:\test
```

然后排除所有这些路径：

```
C:\Program Files
C:\Program Files (x86)
C:\test
C:\test123
```

将排除项从C:\test更改为C:\test\可防止C:\test123被排除。

文件扩展名排除

文件扩展名排除允许排除具有特定扩展名的所有文件。

关键点

- 安全终端控制台中的预期输入为.extension
- 如果未添加任何文件扩展名，安全终端控制台会自动为文件扩展名预置一个时间段。

- 扩展名不区分大小写。

例如，要排除所有Microsoft Access数据库文件，可以创建此排除：

```
.MDB
```



注意：标准文件扩展名例外项在默认列表中可用，不建议删除这些例外项，这样做会导致终端上的性能更改。

通配符排除

通配符例外项与路径或文件扩展名例外项相同，但您可以使用星号字符(*)表示路径或扩展名中的通配符。

例如，如果您要将macOS上的虚拟机排除在扫描范围之外，则可以输入以下路径排除：

```
/Users/johndoe/Documents/Virtual Machines/
```

但是，此排除仅对一个用户有效，因此请用星号替换路径中的用户名，并创建通配符排除以排除所有用户的此目录：

```
/Users/*/Documents/Virtual Machines/
```



警告：通配符排除不会止于路径分隔符，这可能会导致意外排除。例如，C:*\test不包括C:\sample\test和C:\1\test**或C:\sample\test123。



警告：以星号字符开始排除可能会导致严重的性能问题。删除或更改所有以星号字符开头的排除项，以缓解CPU影响。

Windows 窗口版本

在Windows上创建通配符例外项时，有一个选项可以应用到所有驱动器号。选择此选项会将通配符

排除应用到所有已装载的驱动器。



如果您要手动创建相同的排除项，则需要使用^[A-Za-z]对其进行前缀，例如：

```
^[A-Za-z]\testpath
```

两个示例中都排除了C:\testpath和D:\testpath。

如果通配符例外项选择Apply to all drive letters，安全终端控制台会自动生成^[A-Za-z]。

可执行文件排除项（仅限Windows）

可执行排除项仅适用于已启用漏洞防御的Windows连接器。可执行文件排除将某些可执行文件排除在利用漏洞防护的保护范围之外。如果遇到问题或性能问题，则只能将可执行文件从漏洞防御中排除。

您可以检查受保护进程的列表，并通过在应用程序排除字段中指定其可执行名称来排除任何受保护进程的。可执行文件例外项必须与name.exe格式的可执行文件名称完全匹配。不支持通配符。

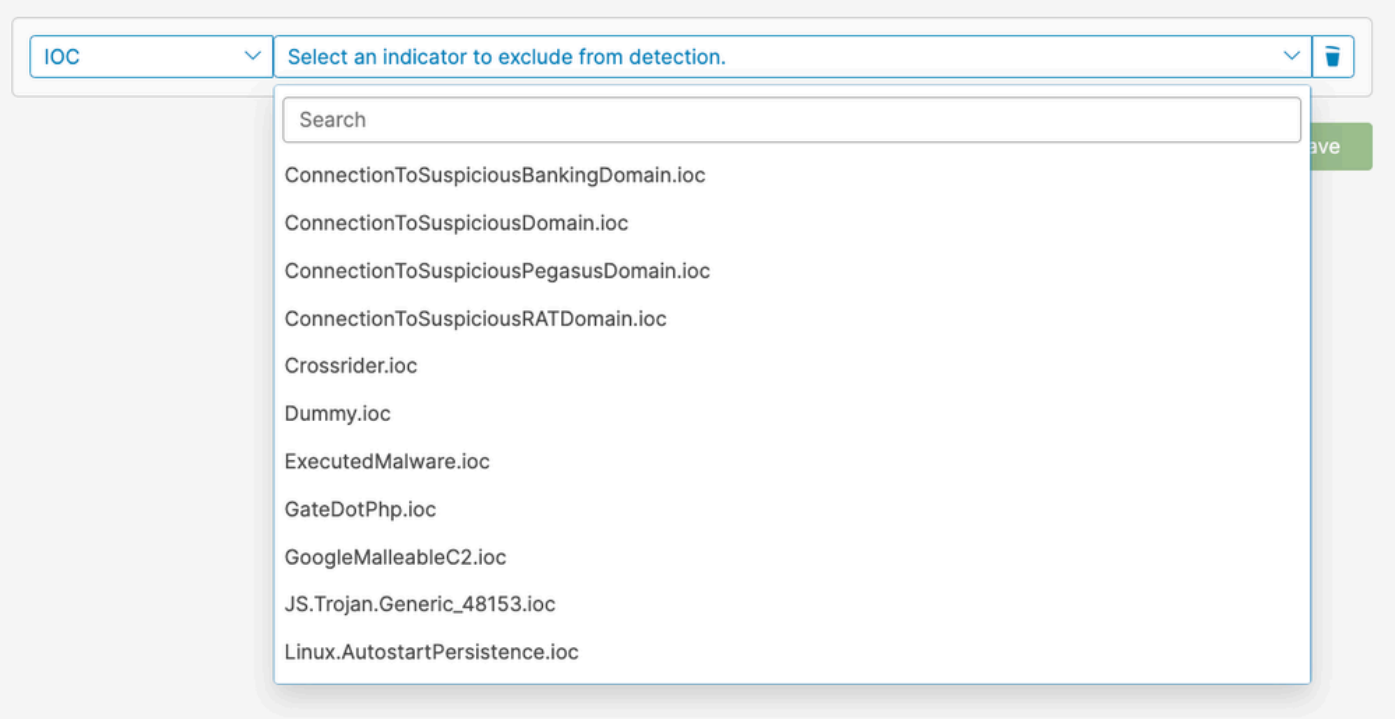
注意：只有应用程序才能通过安全终端控制台使用可执行排除项进行排除。与DLL相关的任何例外项都需要打开一个支持案例，才能创建例外项。

查找正确的漏洞防御排除项是一个比任何其他排除类型要复杂得多的过程，需要进行大量测试以最大限度地减少任何有害的安全漏洞。

IOC排除（仅限Windows）

IOC排除允许排除云危害表现。如果您有一个无法签名的自定义或内部应用程序，导致某些IOC频繁触发，则此功能非常有用。安全终端控制台提供要从IOC排除中选择的指示器列表。

您可以通过下拉列表选择要排除的指标：



注意：如果排除严重性高或严重的IOC，您将无法看到它，并可能使您的组织面临风险。只有在遇到大量误报检测时，才能排除这些IOC。

CSIDL和KNOWFOLDERID（仅Windows）

在编写Windows的路径和进程排除项时，接受并鼓励使用CSIDL和KNOWFOLDERID值。CSIDL/KNOWNFOLDERID值对于为使用备用驱动器号的环境创建进程和路径排除项非常有用。

在使用CSIDL/KNOWNFOLDERID时，需要考虑一些限制。如果您的环境在多个驱动器盘符上安装程序，则CSIDL/KNOWNFOLDERID值仅指标记为默认或已知安装位置的驱动器。

例如，如果操作系统安装在C:\上，但Microsoft SQL的安装路径已手动更改为`D:\`，则维护的排除列表中的基于CSIDL/KNOWNFOLDERID的排除将不适用于该路径。这意味着必须为C:\驱动器上未列出的每个路径或进程排除项输入一个排除项，因为CSIDL/KNOWNFOLDERID的使用不会映射该路径。

有关详细信息，请参阅此Windows文档：

- CSIDL
- KNOWNFOLDERID

注意：仅Windows连接器8.1.7及更高版本支持KNOWNFOLDERID。早期版本的Windows连接器使用CSIDL值。

注意：KNOWNFOLDERID值区分大小写。例如，必须使用值FOLDERID_ProgramFiles，而不能使用无效的valueFolderID_programfiles。

准备用于排除调整的连接器

要准备用于排除调优的连接器，您需要：

1. 设置要在调试模式下运行的策略和组。
2. 根据正常业务操作运行新调试组中的计算机，留出时间获取足够的连接器日志数据。
3. 在连接器上生成诊断数据，用于识别排除项。

有关在不同操作系统上启用调试模式和收集诊断数据的说明，请参阅以下文档：

- 适用于Mac诊断数据收集的思科安全终端连接器
- 用于Linux诊断数据收集的思科安全终端连接器
- 分析高CPU的AMP诊断套件(Windows)

确定排除项

MacOS和Linux

在Debug模式下生成的诊断数据提供了两个文件，可用于创建例外项：fileops.txt和execs.txt。fileops.txt文件用于创建路径/文件扩展名/通配符例外项，而execs.txt文件用于创建进程例外项。

创建进程排除

execs.txt文件列出触发安全终端执行文件扫描的可执行路径。每条路径都有一个关联的计数，该计数指示该路径被扫描的次数，并且列表按降序排序。您可以使用此列表确定执行事件数量较多的进程，然后使用进程路径建立例外项。但是，建议不要排除常规实用程序（如/usr/bin/grep）或解释程序（如/usr/bin/ruby）。

如果通用程序或解释程序生成大量文件扫描，您可以进行更多调查以尝试制定更有针对性的例外项：

1. 排除父进程：确定哪个应用程序正在执行进程（例如，查找执行grep的父进程），并排除此父进程。当且仅当父进程可以安全转换为进程排除时，必须执行此操作。如果父代排除适用于子代，则从父代进程到任何子代的调用也将被排除。
2. 排除给定用户的进程：确定哪个用户正在执行进程。如果某个特定用户正在大量执行进程，您可以仅为该特定用户排除该进程（例如，如果某个进程正被用户root以大量调用时，您可以排除该进程，但只能针对指定的用户root，这允许安全终端监控任何非根用户执行给定进程的情况）。

execs.txt的输出示例：

```
33 /usr/bin/bash
23 /usr/bin/gawk
21 /usr/bin/wc
21 /usr/bin/sleep
21 /usr/bin/ls
19 /usr/bin/pidof
17 /usr/bin/sed
14 /usr/bin/date
13 /usr/libexec/gdb
13 /usr/bin/iconv
11 /usr/bin/cat
10 /usr/bin/systemctl
 9 /usr/bin/pgrep
 9 /usr/bin/kmod
 7 /usr/bin/rm
 6 /usr/lib/systemd/systemd-cgroups-agent
 6 /usr/bin/rpm
 4 /usr/bin/tr
 4 /usr/bin/sort
 4 /usr/bin/find
```

创建路径、文件扩展名和通配符例外项

fileops.txt文件列出文件创建、修改和重命名活动触发安全端点执行文件扫描的路径。每条路径都有一个关联的计数，该计数指示该路径被扫描的次数，并且列表按降序排序。开始使用Path exclusions的方法之一是从fileops.txt中查找最频繁扫描的文件和文件夹路径，然后考虑为这些路径创建规则。虽然高计数不一定表示必须排除路径（例如可以经常扫描存储电子邮件的目录，但决不能排除该目录），但列表提供了识别排除候选目录的起点。

fileops.txt的输出示例：

```
31 /Users/eugene/Library/Cookies/Cookies.binarycookies
24 /Users/eugene/.zhistory
9 /Users/eugene/.vim/.temp/viminfo
9 /Library/Application Support/Apple/ParentalControls/Users/eugene/2018/05/10-usage.data
5 /Users/eugene/Library/Cookies/HSTS.plist
5 /Users/eugene/.vim/.temp/viminfo.tmp
4 /Users/eugene/Library/Metadata/CoreSpotlight/index.spotlightV3/tmp.spotlight.state
3 /Users/eugene/Library/WebKit/com.apple.Safari/WebsiteData/ResourceLoadStatistics/full_browsing_sessio
3 /Library/Logs/Cisco/supporttool.log
2 /private/var/db/locationd/clients.plist
2 /Users/eugene/Desktop/.DS_Store
2 /Users/eugene/.dropbox/instance1/config.dbx
2 /Users/eugene/.DS_Store
2 /Library/Catacomb/DD94912/biolockout.cat
2 /.fseventsd/000000000029d66b
1 /private/var/db/locationd/.dat.nosync0063.arg4tq
```

一个好的经验法则是，任何具有日志或日志文件扩展名的事物都必须被视为合适的排除候选。

行为保护引擎

行为保护引擎在Linux连接器版本1.22.0和macOS连接器版本1.24.0中引入；从这些版本开始，连接器可以检测绝大多数高系统活动，然后引发故障18。

进程排除项应用于所有引擎和文件扫描。将进程排除应用于非常活跃的良性进程，以修复此故障。top.txt文件由调试模式诊断数据生成，可用于确定系统中最活跃的进程。有关详细的补救步骤，请参阅安全终端Mac/Linux连接器故障18指南。

此外，进程排除可以阻止良性软件的误报行为保护检测。对于Secure Endpoint Console中的误报检测，可以排除该进程以改进报告。

Windows 窗口版本

Windows操作系统更加复杂，由于父进程和子进程，有更多排除选项可用。这表示需要进行更深入的审查，以识别已访问的文件以及生成这些文件的程序。

请参阅思科安全解决方案的GitHub页面上的此Windows调整工具，以获取有关如何分析和优化安全终端的Windows性能的详细信息。

在安全终端控制台中创建排除规则

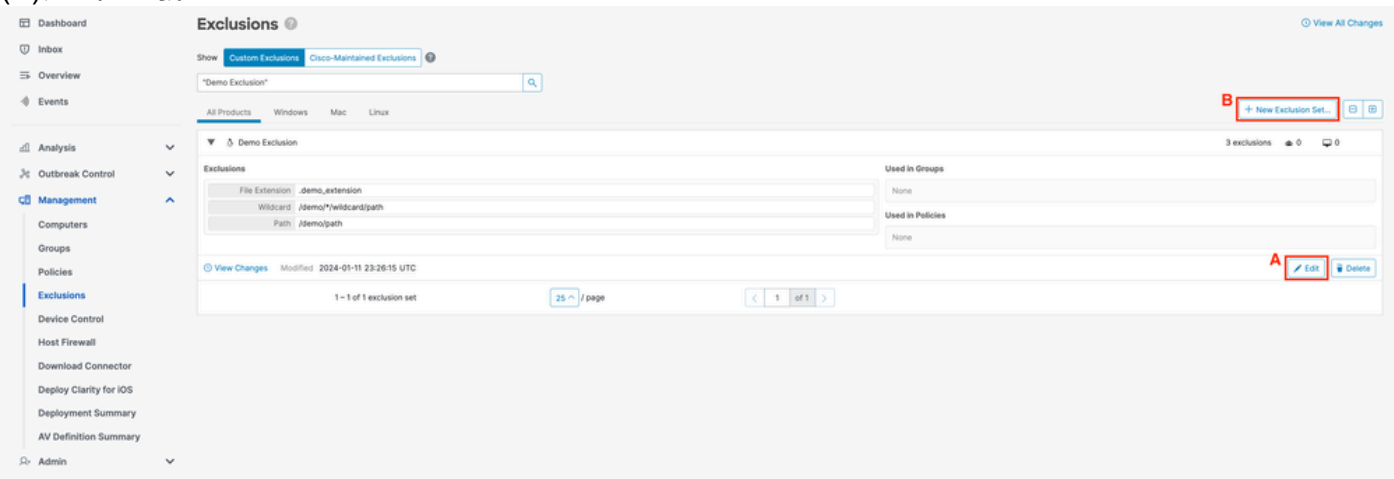
⚠ 警告：在写入例外项之前，请始终了解文件和进程，以避免终端上的安全漏洞。

完成以下步骤，使用安全终端控制台创建新的排除规则：

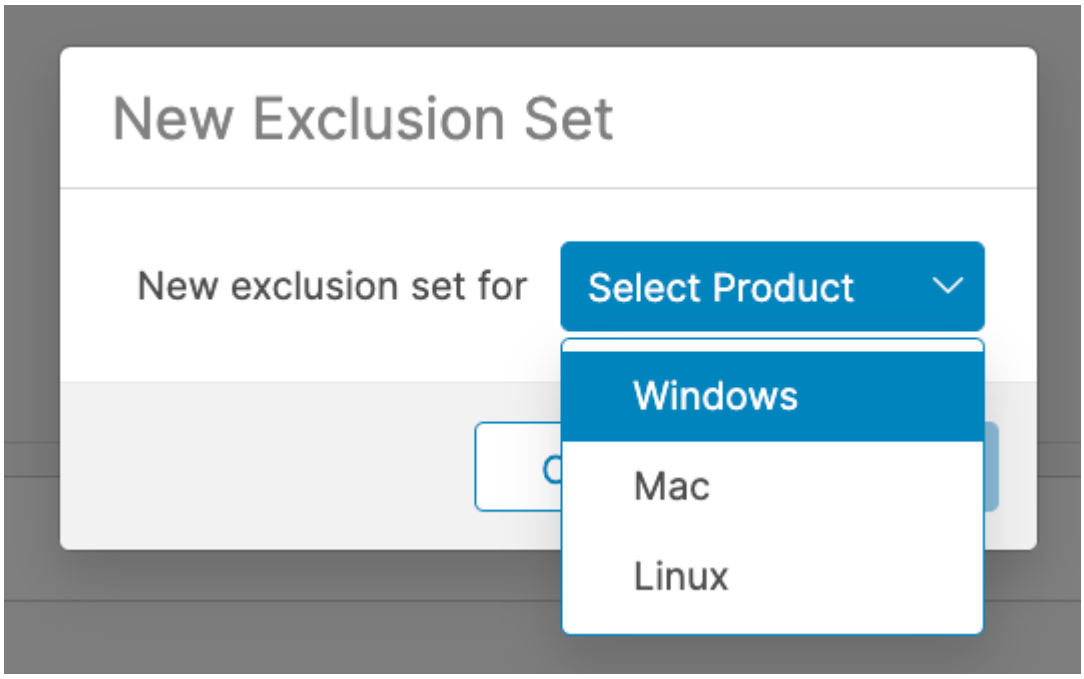
1. 在安全终端控制台中，通过选择管理>排除导航到策略页面。以下任一项：

(A)找到要修改的排除集并单击Edit，或

(B)点击+新排除集.....

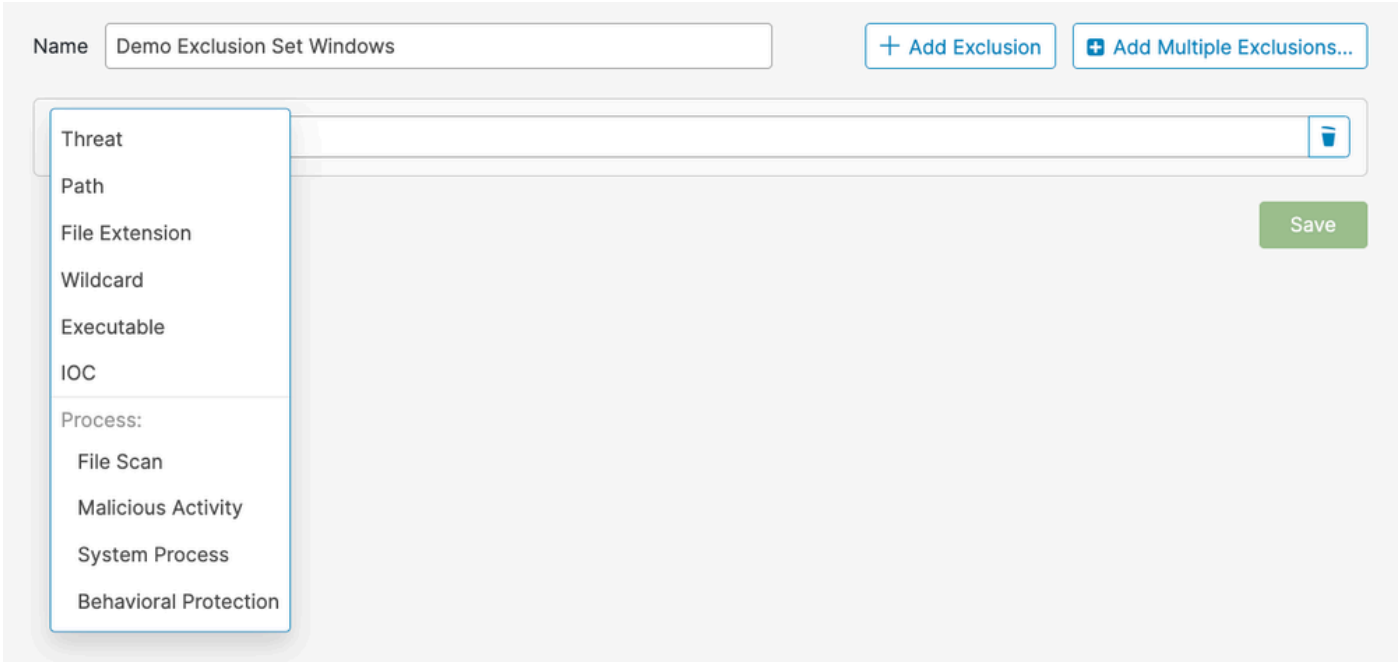


1. 在New Exclusion Set弹出窗口中，选择要为其创建排除集的操作系统。Click Create.

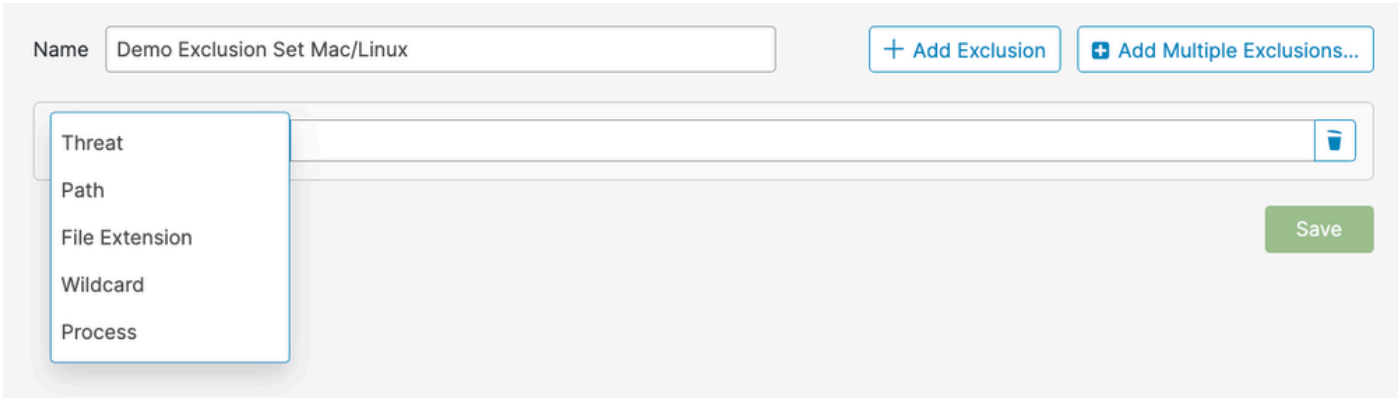


2. 系统会将您重定向到“新的排除集”页面。单击+ Add Exclusion，然后从Select Type下拉列表中选择exclusion类型。

Windows 窗口版本:



Mac/Linux:



1. 填写所选排除类型的必填字段。
2. 重复步骤2和3以添加更多规则，或者单击Save以保存例外项集。

最佳实践

⚠ 警告：创建例外项时应小心，因为它们会降低思科安全终端提供的保护级别。排除的文件未进行散列、扫描或在缓存或云中可用，活动未受监控，并且后端引擎、设备轨迹和高级分析中缺少信息。

排除项只能用于目标实例，如与特定应用程序的兼容性问题，或无法改善的性能问题。

创建例外项时需要遵循的一些最佳实践包括：

- 仅针对已验证的问题创建例外项。
  - 不要认为排除是必需的，除非已经证明这是一个否则无法解决的问题。
  - 应用排除之前，必须彻底调查和缓解性能问题、误报或应用程序兼容性问题。
- 首选进程例外项而不是路径/文件扩展名/通配符例外项。
  - 与结合使用路径、文件扩展名和通配符排除来达到相同结果相比，进程排除提供了一种更直接的方式来排除良性软件活动。
  - 建议尽可能使用相应的进程排除项替换目标程序可执行文件的路径、文件扩展名和通配符排除项。
- 避免广泛排除。
  - 不要排除终端的大部分内容，例如整个C驱动器。
  - 使用文件的完全限定路径，而不只是文件名。
  - 使用设备轨迹、安全终端诊断数据和Windows调整工具来调查和确定特定排除项。
- 避免过度使用通配符排除项。
  - 使用通配符创建例外项时请小心。尽可能使用更具体的排除项。
  - 在排除中使用最少量的通配符；只有真正可变的文件夹必须使用通配符。
- 避免排除通用程序和口译员。
  - 建议不要排除通用程序或解释程序。
  - 如果确实需要排除排除的常规实用程序或解释程序，则提供一个进程用户（仅限macOS/Linux）。
  - 例如，避免写入包括python、java、ruby、bash和sh的例外项。
- 避免重复排除。
  - 创建排除之前，请检查该排除是否已存在于自定义排除项或思科维护的排除项中。
  - 删除重复排除项可提高性能并减少排除项的操作管理。
  - 确保路径/文件扩展名/通配符排除未涵盖进程排除中指定的路径。
- 避免排除已知常用于恶意软件攻击的进程。
  - 有关详细信息，请参阅不推荐的排除。
- .删除过时的排除项。
  - 定期检查和审核您的排除列表，并记录添加某些排除的原因。
- 删除危害时的排除项。
  - 当连接器受到威胁时，必须删除例外项，以便重新获得最佳安全性和可视性。
  - 自动操作可用于在感染后对连接器应用更安全的策略。如果连接器受损，则必须将其移

至包含策略的组，且不得有任何例外情况，以确保应用最高级别的保护。
    - 有关如何主动设置“危害时移动计算机到组”自动化操作的详细信息，请参阅确定在安全终端中触发自动化操作的条件。
- 加强对排除项目的保护。
    - 当排除项绝对有必要时，请考虑可以采取什么缓解策略，例如启用写保护以便为排除项添加一些保护层。
- 智能地创建例外项。
    - 通过选择最高级别的父进程来优化规则，该父进程可唯一标识要排除的应用程序，并使用“应用到子进程”选项来最小化规则数。
- 切勿排除启动过程。
    - 启动进程（在macOS上启动，在Linux上初始化或系统启动）负责启动系统上的所有其它进程，并且位于进程层次结构的顶部。
    - 排除启动进程及其所有子进程将有效禁用安全终端监控。
- 尽可能指定进程用户（仅限macOS/Linux）。
    - 如果用户字段留空，则排除项适用于运行指定程序的任何进程。
    - 虽然适用于任何用户的例外项更具灵活性，但此广泛范围可能会无意中排除必须监控的活动。
    - 指定用户对于应用于运行时引擎（例如java）和脚本解释程序（例如bash和python）等共享程序的规则尤其重要。
    - 指定用户可限制范围，并指示安全终端在监控其他实例时忽略特定实例。

不推荐的排除项

虽然不可能知道攻击者可以使用的所有可能的攻击媒介，但必须监控一些核心攻击媒介。为了保持良好的安全状态和可视性，不建议使用以下排除项：

AcroRd32.exe
addinprocess.exe
addinprocess32.exe
addinutil.exe
bash.exe
bginfo.exe
bitsadmin.exe
cdb.exe
csi.exe
dbghost.exe
dbgsvc.exe
dnx.exe
dotnet.exe
excel.exe
fsi.exe
fsiAnyCpu.exe
iexplore.exe
java.exe

kd.exe
lxssmanager.dll
msbuild.exe
mshta.exe
ntkd.exe
ntsd.exe
outlook.exe
psexec.exe
powerpnt.exe
powershell.exe
rcsi.exe
svchost.exe
schtasks.exe
system.management.automation.dll
windbg.exe
winword.exe
wmic.exe
wuault.exe
.7z
.bat
.bin
.cab
.cmd
.com
.cpl
.dll
.exe
.fla
.gif
.gz
.hta
.inf
.java
.jar
.job
.jpeg
.jpg
.js
.ko
.ko.gz
.msi
.ocx

.png
.ps1
.py
.rar
.reg
.scr
.sys
.tar
.tmp
.url
.vbe
.vbs
.wsf
.zip
bash
Java
python
python3
sh
zsh
/
/bin
/sbin
/usr/lib
C :
C:\
C:*
D:\
D:*
C:\Program Files\Java
C:\Temp\
C:\Temp*
C:\Users\
C:\Users*
C:\Windows\Prefetch
C:\Windows\Prefetch\
C:\Windows\Prefetch*
C:\Windows\System32\Spool
C:\Windows\System32\CatRoot2
C:\Windows\Temp
C:\Windows\Temp\

C:\Windows\Temp*
C:\Program文件\<公司名称>\
C:\Program文件(x86)\<公司名称>\
C:\Users\<UserProfileName>\AppData\Local\Temp\
C:\Users\<UserProfileName>\AppData\LocalLow\Temp\



注意：这不是要避免的详尽排除列表，但可以深入了解核心攻击媒介。保持对这些路径、文件扩展名和进程的可视性至关重要。

相关信息

- 技术支持和文档 - Cisco Systems
- 思科安全终端 — TechNote
- 思科安全终端 — 用户指南
- 排除安全终端中的漏洞防御故障
- 确定触发安全终端中自动操作的条件

关于此翻译
思科采用人工翻译与机器翻译相结合的方式将此文档翻译成不同语言，希望全球的用户都能通过各自的语言得到支持性的内容。

请注意：即使是最好的机器翻译，其准确度也不及专业翻译人员的水平。

Cisco Systems, Inc. 对于翻译的准确性不承担任何责任，并建议您总是参考英文原始文档（已提供链接）。