System Center Endpoint Protection

Installations- und Benutzerhandbuch

Red Hat Enterprise Linux Server 5, 6
SUSE Linux Enterprise 10, 11
CentOS 5, 6
Debian Linux 5, 6
Ubuntu Linux 10.04, 12.04
Oracle Linux 5, 6

Microsoft

Microsoft System Center Endpoint Protection

Inhalt

Einführung

Vielen Dank, dass Sie sich für System Center Endpoint Protection entschieden haben. Das moderne Microsoft-Prüfmodul dieser Lösung überzeugt durch eine sehr hohe Prüfgeschwindigkeit und Erkennungsrate. In Verbindung mit dem sehr geringen Ressourcenbedarf eignet sich dieses System daher ideal zum Absichern von Linux-Servern.

Hauptfunktionen

On-Demand-Prüfung

Die On-Demand-Prüfung kann durch einen Benutzer mit entsprechenden Rechten (in der Regel Systemadministratoren) über die Kommandozeilen-Schnittstelle, die Web-Oberfläche oder den Taskplaner des Betriebssystems (z. B. cron) gestartet werden. *On-Demand* heißt diese Prüfung, weil Dateisystem-Objekte dabei auf Anforderung („on demand“) des Benutzers oder des Systems geprüft werden.

Echtzeit-Schutz

Der Echtzeit-Schutz tritt dann in Aktion, wenn ein Benutzer oder das Betriebssystem versucht, auf Dateisystem-Objekte zuzugreifen. *Echtzeit* bedeutet hier also, dass die Prüfung unmittelbar beim Zugriff auf diese Objekte erfolgt.

Hauptmerkmale des Systems

Hochentwickelte Prüfalgorithmen

Die Algorithmen des Microsoft-Virenschutz-Prüfmoduls überzeugen durch eine maximale Erkennungsrate bei gleichzeitig sehr hoher Prüfgeschwindigkeit.

Unterstützung von Mehrprozessor-Systemen

System Center Endpoint Protection wurde für die Ausführung auf Einzel- und Mehrprozessor-Systemen entwickelt.

Advanced Heuristik

System Center Endpoint Protection bietet eine spezielle „Advanced Heuristik“ für Win32-Würmer, Backdoor-Infektionen und andere Malware-Formen.

Integrierte Archivfunktionen

Durch integrierte Routinen für die Verarbeitung von Archiven können Archivdateien ohne externe Zusatzprogramme entpackt werden.

Geschwindigkeit und Effizienz

Für maximale Geschwindigkeit und Effizienz basiert die Architektur von System Center Endpoint Protection auf einem im Hintergrund ausgeführten Daemon, an den alle Prüfanforderungen gesendet werden.

Höhere Sicherheit

Alle ausgeführten Daemons (außer scep_dac) werden unter einem eingeschränkten Benutzerkonto ausgeführt, um Sicherheitsrisiken zu vermeiden.

Selektive Konfiguration

Das System kann selektiv für einzelne Benutzer oder Clients/Server konfiguriert werden.

Mehrere Logging-Stufen

Informationen zur Systemaktivität und zu Infiltrationen können über mehrere konfigurierbare Logging-Stufen protokolliert werden.

Web-Oberfläche

Die Konfiguration und Administration ist über eine intuitiv bedienbare, benutzerfreundliche Web-Oberfläche möglich.

Keine externen Bibliotheken

Zur Installation von System Center Endpoint Protection sind keine externen Bibliotheken oder Programme außer libc erforderlich.

Gezielte Benachrichtigung bestimmter Benutzer

Das System kann so konfiguriert werden, dass bei einer erkannten Infiltration oder anderen wichtigen Ereignissen bestimmte Benutzer benachrichtigt werden.

Geringe Ressourcenanforderungen

Ein effizienter Betrieb von System Center Endpoint Protection ist schon mit 16 MB Festplattenspeicher und 32 MB RAM möglich. Das System unterstützt die Linux-Kernelversionen 2.2.x, 2.4.x und 2.6.x.

Performance und Skalierbarkeit

Wie man es von einer Unix-basierten Lösung erwartet, bietet System Center Endpoint Protection hohe Performance und Skalierbarkeit für Rechnerumgebungen aller Größenklassen – von kleineren Büroservern bis hin zu großen ISP-Servern mit Tausenden von Benutzern. Gleichzeitig profitieren Sie vom hervorragenden Schutzniveau einer Microsoft-Sicherheitslösung.

Begriffe und Abkürzungen

In diesem Abschnitt sind die im vorliegenden Handbuch verwendeten Begriffe und Abkürzungen beschrieben. Fettdruck kennzeichnet die Namen von Produktkomponenten sowie erstmals erwähnte Begriffe und Abkürzungen. Die in diesem Kapitel definierten Begriffe und Abkürzungen werden im Rest dieses Dokuments näher erläutert und beschrieben.

SCEP

SCEP ist die Abkürzung für ein von Microsoft entwickeltes Security-Produkt für Linux-Betriebssysteme. Gleichzeitig heißt so das Softwarepaket, das diese Produkte enthält.

SCEP daemon

Der zentrale SCEP-Daemon, der das System steuert und Prüfaufgaben übernimmt: *scep_daemon*.

SCEP-Basisverzeichnis

Das Verzeichnis, in dem die SCEP-Lademodule mit der Signaturdatenbank abgelegt sind. Für dieses Verzeichnis wird in diesem Dokument die Abkürzung *@BASEDIR@* verwendet. Je nach Betriebssystem lautet *@BASEDIR@* wie folgt:

```
Linux: /var/opt/microsoft/scep/lib
```

SCEP-Konfigurationsverzeichnis

Das Verzeichnis, in dem alle Konfigurationsdateien für System Center Endpoint Protection gespeichert werden. Für dieses Verzeichnis wird in diesem Dokument die Abkürzung *@ETCDIR@* verwendet. Je nach Betriebssystem lautet *@ETCDIR@* wie folgt:

```
Linux: /etc/opt/microsoft/scep
```

SCEP-Konfigurationsdatei

Die zentrale Konfigurationsdatei für System Center Endpoint Protection. Der absolute Pfad zu dieser Datei lautet:

@ETCDIR@/scep.cfg

SCEP-Binärdateiverzeichnis

Das Verzeichnis, in dem die Binärdateien für System Center Endpoint Protection abgelegt sind. Für dieses Verzeichnis wird in diesem Dokument die Abkürzung *@BINDIR@* verwendet. Je nach Betriebssystem lautet *@BINDIR@* wie folgt:

```
Linux: /opt/microsoft/scep/bin
```

SCEP-Systembinärdateiverzeichnis

Das Verzeichnis, in dem die Systembinärdateien für System Center Endpoint Protection abgelegt sind. Für dieses Verzeichnis wird in diesem Dokument die Abkürzung *@SBINDIR@* verwendet. Je nach Betriebssystem lautet *@SBINDIR@* wie folgt:

```
Linux: /opt/microsoft/scep/sbin
```

SCEP-Objektdateiverzeichnis

Das Verzeichnis, in dem die Objektdateien und Bibliotheken für System Center Endpoint Protection abgelegt sind. Für dieses Verzeichnis wird in diesem Dokument die Abkürzung *@LIBDIR@* verwendet. Je nach Betriebssystem lautet *@LIBDIR@* wie folgt:

```
Linux: /opt/microsoft/scep/lib
```

Installation

System Center Endpoint Protection wird in Form einer Binärdatei bereitgestellt:

```
scep.i386.ext.bin
```

Im oben gezeigten Dateinamen steht *'ext'* für ein Suffix, das angibt, für welche Linux-Distribution sich die Datei eignet: „deb“ für Debian, „rpm“ für RedHat und SuSE oder „tgz“ für andere Linux-Distributionen.

Um das Produkt oder ein Upgrade dafür zu installieren, verwenden Sie den folgenden Befehl:

```
sh ./scep.i386.ext.bin
```

Hiermit wird zunächst die Lizenzvereinbarung für das Produkt angezeigt. Nachdem Sie diese akzeptiert haben, wird das Installationspaket im aktuellen Arbeitsverzeichnis abgelegt. Anschließend werden Sie am Bildschirm durch die (De-)Installation bzw. den Upgrade-Vorgang geführt.

Nach der Installation des Pakets können Sie mit dem folgenden Befehl überprüfen, ob der primäre SCEP-Dienst ausgeführt wird:

```
ps -C scep_daemon
```

Nach Bestätigung mit der Eingabetaste sollten Sie eine Meldung wie die folgende sehen:

```
 PID TTY TIME CMD
2226 ? 00:00:00 scep_daemon
2229 ? 00:00:00 scep_daemon
```

Es werden mindestens zwei SCEP-Daemon-Prozesse im Hintergrund ausgeführt. Die erste PID steht für den Prozess- und Thread-Manager des Systems. Bei der zweiten PID handelt es sich um den Prüfprozess.

Installieren eines Sprachpakets

Zur Installation des erforderlichen Sprachpakets für System Center Endpoint Protection verwenden Sie den folgenden Befehl:

```
sh ./scep-lang.lng.bin
```

Dabei ersetzen Sie *'lng'* durch den Sprachcode der zu importierenden Datei.

Nach der Meldung *Installation completed successfully* müssen Sie nun noch die Systemvariable LANG anpassen und ggf. die Umgebung aktualisieren. Die Installation des Sprachpakets ist damit abgeschlossen.

Sprachpakete enthalten jeweils die folgenden Komponenten:

- Übersetzte Web-Oberfläche
- Übersetzte Konsolenausgabe von SCEP-Agents und -Befehlen
- Übersetzte PDF-Dokumentation

Architekturübersicht

Nach der Installation von System Center Endpoint Protection sollten Sie sich kurz mit der Architektur dieses Systems vertraut machen.

Das System besteht aus den folgenden Teilen:

KERN

Den Kern von System Center Endpoint Protection stellt der SCEP-Daemon (scep_daemon) dar. Auf Grundlage der SCEP-API-Bibliothek libscep.so und den SCEP-Lademodulen em00X_xx.dat stellt er Kernfunktionen wie das Prüfen von Inhalten, die Verwaltung der Agent-Daemon-Prozesse sowie des Systems zum Einreichen von Malware-Proben, das Erstellen der Logs oder Notifikationen bereit. Ausführlichere Informationen finden Sie in der Manpage *scep_daemon(8)*.

AGENTS

Die Agent-Module von SCEP dienen zur Integration von SCEP mit der Linux-Serverumgebung.

DIENSTPROGRAMME

Die Dienstprogramm-Module ermöglichen eine einfache, effektive Systemverwaltung. Sie decken Systemaufgaben wie die Quarantäneverwaltung, die Einrichtung des Systems sowie Updates ab.

KONFIGURATION

Eine korrekte Konfiguration ist der wichtigste Aspekt eines Sicherheitssystems. Der Rest dieses Kapitels widmet sich daher den diesbezüglichen Komponenten. Zudem sollten Sie sich etwas gründlicher mit der Datei *scep.cfg* beschäftigen, da sie wichtige Informationen zur Konfiguration von System Center Endpoint Protection enthält.

Nach Abschluss der Installation werden alle Konfigurationskomponenten des Produkts im SCEP-Konfigurationsverzeichnis gespeichert. Dieses besteht aus den folgenden Dateien:

@ETCDIR@/scep.cfg

Diese Datei ist die wichtigste Konfigurationsdatei und steuert alle wesentlichen Aspekte der Produktfunktionalität. Sie besteht aus mehreren Abschnitten mit jeweils einer Reihe von Parametern. Neben einem Abschnitt „global“ mit globalen Parametern gibt es mehrere Abschnitte zu den unterschiedlichen Agents. Die Abschnittnamen sind daran zu erkennen, dass sie in eckigen Klammern stehen. Die Parameter im Abschnitt „global“ dienen zur Definition der Konfigurationsoptionen für den SCEP-Daemon sowie der Standardwerte für die Konfiguration des SCEP-Prüfmoduls. Die Parameter in den Abschnitten zu den einzelnen Agents definieren die Konfigurationsoptionen der Module, mit denen verschiedene Arten von Datenströmen auf dem Computer und im Netzwerk abgefangen und geprüft werden. Neben den diversen Parametern zur Systemkonfiguration gibt es auch eine Reihe von Regeln für den Aufbau der Datei. Ausführliche Informationen, wie Sie diesen Aufbau möglichst effektiv gestalten, finden Sie in den Manpages *scep.cfg(5)* und *scep_daemon(8)* sowie in den Manpages zu den einzelnen Agents.

@ETCDIR@/certs

In diesem Verzeichnis werden die Zertifikate zur Authentifizierung in der Web-Oberfläche von SCEP gespeichert. Weitere Informationen finden Sie in der Manpage *scep_wwwi(8)*.

@ETCDIR@/scripts/daemon_notification_script

Sofern über den Parameter *'exec_script'* in der SCEP-Konfigurationsdatei aktiviert, wird dieses Skript ausgeführt, wenn der Virenschutz eine Infiltration erkennt. Es sendet eine E-Mail mit Informationen zu der erkannten Bedrohung an den Systemadministrator.

Integration mit Dateisystem-Diensten

In diesem Kapitel wird beschrieben, wie Sie die On-Demand-Prüfung und den Echtzeit-Schutz so konfigurieren, dass das Dateisystem effektiv vor Infektionen durch Viren und Würmer geschützt wird. Auf die Prüffunktionen von System Center Endpoint Protection greifen Sie mit den Befehlen *scep_scan* (On-Demand-Prüfung) sowie *scep_dac* (Echtzeit-Schutz) zu. Die Linux-Version von System Center Endpoint Protection bietet darüber hinaus eine zusätzliche Methode für den Echtzeit-Schutz auf Basis der Preload-Bibliothek *libscep_pac.so*. Diese Befehle werden in den nachfolgenden Abschnitten beschrieben.

On-Demand-Prüfung

Die On-Demand-Prüfung kann durch einen Benutzer mit entsprechenden Rechten (in der Regel Systemadministratoren) über die Kommandozeilen-Schnittstelle, die Web-Oberfläche oder den Taskplaner des Betriebssystems (z. B. cron) gestartet werden. *On-Demand* heißt diese Prüfung, weil Dateisystem-Objekte dabei auf Anforderung („on demand“) des Benutzers oder des Systems geprüft werden.

Zum Durchführen einer On-Demand-Prüfung ist keine besondere Konfiguration erforderlich. Die On-Demand-Prüfung kann sofort über die Kommandozeile oder den Taskplaner gestartet werden, nachdem das SCEP-Paket korrekt installiert wurde. Zum Starten der On-Demand-Prüfung aus der Kommandozeile verwenden Sie die folgende Syntax:

```
@SBINDIR@/scep_scan [option(s)] FILES
```

FILES ist dabei eine Liste der Verzeichnisse und/oder Dateien, die geprüft werden sollen.

Die On-Demand-Prüfung von SCEP unterstützt eine Reihe von Kommandozeilen-Optionen. Eine vollständige Liste finden Sie in der Manpage *scep_scan(8)*.

Echtzeit-Schutz auf Basis von Dazuko

Der Echtzeit-Schutz tritt dann in Aktion, wenn ein Benutzer oder das Betriebssystem versucht, auf Dateisystem-Objekte zuzugreifen. *Echtzeit* bedeutet hier also, dass die Prüfung unmittelbar beim Zugriff auf diese Objekte erfolgt.

Der Echtzeit-Schutz von SCEP basiert auf dem Kernelmodul Dazuko, über das die für den Zugriff erforderlichen Kernelaufrufe abgefangen werden. Dazuko ist ein Open-Source-Projekt, sein Quellcode ist also frei zugänglich. Das Kernelmodul kann dadurch auch für individuell angepasste Kernels kompiliert werden. Da das Dazuko-Kernelmodul als separates Projekt nicht Bestandteil der SCEP-Produkte ist, muss es zunächst kompiliert und im Kernel installiert werden, bevor der Echtzeit-Schutz über den Befehl *scep_dac* verwendet wird. Die Verwendung von Dazuko hat den Vorteil, dass der Echtzeit-Schutz dadurch unabhängig vom verwendeten Dateisystem-Typ wird. Außerdem ermöglicht dieser Ansatz das Prüfen von Dateisystem-Objekten über Network File System (NFS), Nettalk und Samba.

Wichtig: Beachten Sie bitte, dass der Echtzeit-Schutz primär für extern eingehängte Dateisysteme entwickelt und getestet wurde. Falls mehrere Dateisysteme vorhanden sind, bei denen es sich jedoch nicht um extern eingehängte Dateisysteme handelt, müssen diese von der Zugriffskontrolle ausgeschlossen werden, um Systemabstürze zu vermeiden. Typische Beispiele für solche Verzeichnisse sind */dev* und die von SCEP verwendeten Verzeichnisse.

Funktionsprinzip

Der Echtzeit-Schutz *scep_dac* (SCEP Dazuko-powered file Access Controller) ist ein Hintergrundprogramm zur laufenden Überwachung und Kontrolle des Dateisystems. Dabei wird jedes Dateisystem-Objekt bei bestimmten, individuell einstellbaren Zugriffsarten geprüft. In der aktuellen Version werden die folgenden Zugriffsarten unterstützt:

Öffnen

Um die Prüfung bei dieser Zugriffsart zu aktivieren, setzen Sie den Parameter *'event_mask'* im Abschnitt **[fac]** der Datei scep.cfg auf den Wert „open“. Hiermit wird das Bit ON_OPEN der Dazuko-Zugriffsmaske aktiviert.

Schließen

Um die Prüfung bei dieser Zugriffsart zu aktivieren, setzen Sie den Parameter *'event_mask'* im Abschnitt **[fac]** der Datei scep.cfg auf den Wert „close“. Hiermit wird das Bit ON_OPEN der Dazuko-Zugriffsmaske aktiviert. Hiermit werden die Bits ON_CLOSE und ON_CLOSE_MODIFIED der Dazuko-Zugriffsmaske aktiviert.

Hinweis: In bestimmten Kernelversionen können Ereignisse des Typs ON_CLOSE nicht abgefangen werden. In solchen Fällen werden diese Ereignisse von *scep_dac* nicht überwacht.

Ausführen

Um die Prüfung bei dieser Zugriffsart zu aktivieren, setzen Sie den Parameter *'event_mask'* im Abschnitt **[fac]** der Datei scep.cfg auf den Wert „exec“. Hiermit wird das Bit ON_EXEC der Dazuko-Zugriffsmaske aktiviert.

Mit dem Echtzeit-Schutz lässt sich auf diese Weise gewährleisten, dass Dateien beim Öffnen, Schließen oder Ausführen zunächst durch *scep_daemon* geprüft werden. Je nach dem Ergebnis der Prüfung wird der Zugriff auf die jeweilige Datei dann entweder verweigert oder gewährt.

Installation und Konfiguration

Vor der Initialisierung von *scep_dac* muss das Dazuko-Kernelmodul kompiliert und im verwendeten Kernel installiert werden. Weitere Informationen zum Kompilieren und Installieren von Dazuko finden Sie unter folgender Adresse:

http://www.dazuko.org

Nach der Installation von Dazuko müssen Sie die Abschnitte **[global]** und **[fac]** der SCEP-Konfigurationsdatei (scep.cfg) überprüfen und bearbeiten. Insbesondere die korrekte Funktion des Echtzeit-Schutzes hängt von der Konfiguration der Option *'agent_type'* im Abschnitt **[fac]** dieser Datei ab. Zusätzlich müssen Sie definieren, welche Dateisystem-Objekte (also Verzeichnisse und Dateien) vom Echtzeit-Schutz überwacht werden sollen. Dies erfolgt über die Parameter der Optionen *ctl_incl* und *ctl_excl*, die sich ebenfalls im Abschnitt **[fac]** befinden. Nach dem Bearbeiten der Datei scep.cfg können Sie durch einen Neustart des SCEP-Daemons bewirken, dass die neu erstellte Konfiguration eingelesen wird.

Tipps

Um sicherzustellen, dass das Dazuko-Modul vor der Initialisierung des *scep_dac*-Daemons geladen wird, gehen Sie wie folgt vor:

Kopieren Sie das Dazuko-Modul in eines der folgenden, für Kernelmodule reservierten Verzeichnisse:

```
/lib/modules
```

oder

```
/modules
```

Stellen Sie mit den Kernel-Dienstprogrammen 'depmod' und 'modprobe' (unter BSD: 'kldconfig' und 'kldload') sicher, dass alle Abhängigkeiten korrekt ermittelt werden und das neu hinzugefügte Dazuko-Modul erfolgreich initialisiert wird.

Fügen Sie im scep_daemon-Initialisierungsskript '/etc/init.d/scep_daemon' vor der Anweisung zum Initialisieren des Daemons die folgende Zeile ein:

```
/sbin/modprobe dazuko
```

Unter BSD muss die Zeile

```
/sbin/kldconfig dazuko
```

in das Skript '/usr/local/etc/rc.d/scep_daemon.sh' eingefügt werden.

Warnung! Die Schritte müssen unbedingt genau in der angegebenen Reihenfolge ausgeführt werden. Wenn sich das Kernelmodul nicht im Kernelmodul-Verzeichnis befindet, kann es nicht korrekt geladen werden, was zu einem Systemabsturz führt.

Echtzeit-Schutz über Preload-Bibliothek für libc

In den vorangegangenen Abschnitten wurde die Integration des Dazuko-basierten Echtzeit-Schutzes in die Dateisystem-Dienste eines Linux- oder BSD-Betriebssystems beschrieben. In einigen Fällen ist diese Lösung jedoch nicht praktikabel. Dies betrifft insbesondere kritische Systeme, bei denen

- der Quellcode und/oder die Konfigurationsdateien für den verwendeten Kernel nicht verfügbar sind,
- der Kernel eher monolithisch als modular aufgebaut ist
- oder das Dazuko-Modul das jeweilige Betriebssystem ganz einfach nicht unterstützt.

In derartigen Fällen sollte der alternative Echtzeit-Schutz verwendet werden, der auf einer Preload-Bibliothek für libc basiert. Dieser ist in den nachfolgenden Abschnitten näher beschrieben. Sie sind nur für Linux-Benutzer relevant und enthalten Informationen zur Verwendung, Installation und Konfiguration des Echtzeit-Schutzes auf Basis der Preload-Bibliothek *'libscep_pac.so'*.

Funktionsprinzip

Die Echtzeit-Schutz-Bibliothek *libscep_pac.so* (SCEP Preload library based file Access Controller) ist eine gemeinsam genutzte Bibliothek, die beim Systemstart aktiviert wird. Sie fängt libc-Aufrufe durch Dateisystem-Server (FTP-Server, Samba-Server usw.) ab und bewirkt, dass jedes Dateisystem-Objekt bei bestimmten, individuell anpassbaren Zugriffsarten geprüft wird. In der aktuellen Version werden die folgenden Zugriffsarten unterstützt:

Öffnen

Diese Zugriffsart wird aktiviert, wenn der Wert des Parameters *'event_mask'* in der Datei esest.cfg (Abschnitt **[fac]**) das Wort *'open'* enthält.

Schließen

Diese Zugriffsart wird aktiviert, wenn der Wert des Parameters *'event_mask'* in der Datei scep.cfg (Abschnitt **[fac]**) das Wort *'close'* enthält. In diesem Fall werden alle Aufrufe von libc-Funktionen zum Schließen von Dateien über Dateideskriptoren und FILE-Streams abgefangen.

Ausführen

Diese Zugriffsart wird aktiviert, wenn der Wert des Parameters *'event_mask'* in der Datei scep.cfg (Abschnitt **[fac]**) das Wort *'exec'* enthält. In diesem Fall werden alle Aufrufe von exec-Funktionen aus der libc abgefangen.

So kann bewirkt werden, dass alle Dateien beim Öffnen, Schließen und Ausführen vom SCEP-Daemon auf Viren geprüft werden. Je nach dem Ergebnis der Prüfung wird der Zugriff auf die jeweilige Datei dann entweder verweigert oder gewährt.

Installation und Konfiguration

Das Bibliotheksmodul *libscep_pac.so* wird über den Standardmechanismus für Preload-Bibliotheken installiert. Dazu müssen Sie in der Umgebungsvariablen *'LD_PRELOAD'* den absoluten Pfad zur Bibliothek *libscep_pac.so* definieren. Weitere Informationen finden Sie in der Manpage *ld.so(8)*.

Hinweis: Die Umgebungsvariable *'LD_PRELOAD'* sollte nur für die Daemon-Prozesse der Netzwerk-Server definiert werden, die vom Echtzeit-Schutz überwacht werden sollen (FTP, Samba usw.). Es wird nicht empfohlen, sämtliche libc-Aufrufe aller Betriebssystemprozesse auf diese Weise abzufangen, da dies zu drastischen Leistungseinbrüchen oder sogar Systemabstürzen führen kann. Daher sollte weder die Datei '/etc/ld.so.preload' verwendet noch die Umgebungsvariable 'LD_PRELOAD' global exportiert werden. Dies würde in beiden Fällen dazu führen, dass alle libc-Aufrufe abgefangen würden, was zu Systemabstürzen bei der Initialisierung führen kann.

Um sicherzustellen, dass nur die gewünschten Dateizugriffe in einem bestimmten Dateisystem abgefangen werden, kann die betreffende Anwendung nach dem folgenden Muster gestartet werden:

```
LD_PRELOAD=@LIBDIR@/libscep_pac.so COMMAND COMMAND-ARGUMENTS
```

'COMMAND COMMAND-ARGUMENTS' steht dabei für den ursprünglichen Ausführungsbefehl.

Außerdem müssen Sie die Abschnitte **[global]** und **[fac]** der SCEP-Konfigurationsdatei (scep.cfg) überprüfen und bearbeiten. Damit der Echtzeit-Schutz korrekt funktioniert, müssen Sie definieren, welche Dateisystem-Objekte (Verzeichnisse und Dateien) mithilfe der Preload-Bibliothek überwacht werden sollen. Hierzu verwenden Sie die Parameter der Optionen *ctl_incl* und *ctl_excl* im Abschnitt **[fac]** der SCEP-Konfigurationsdatei. Nach dem Bearbeiten der Datei scep.cfg können Sie durch einen Neustart des SCEP-Daemons bewirken, dass die neu erstellte Konfiguration eingelesen wird.

Tipps

Damit der Echtzeit-Schutz unmittelbar nach dem Start des Dateisystems aktiviert wird, muss die Umgebungsvariable *'LD_PRELOAD'* im Initialisierungsskript des jeweiligen Netzwerk-Dateiservers definiert werden.

Beispiel: Nehmen wir an, der Echtzeit-Schutz soll alle Zugriffe im Dateisystem unmittelbar nach dem Start des Samba-Servers überwachen. Hierzu müsste im Initialisierungsskript des Samba-Daemons (/etc/init.d/smb) die Anweisung

```
daemon /usr/sbin/smbd $SMBDOPTIONS
```

durch die folgende Zeile ersetzt werden:

```
LD_PRELOAD=@LIBDIR@/libscep_pac.so daemon /usr/sbin/smbd $SMBDOPTIONS
```

Dies bewirkt, dass die von Samba gesteuerten Dateisystem-Objekte nach dem Systemstart geprüft werden.

Wichtige Mechanismen in SCEP

Policy zur Objektverarbeitung

Über die Policy zur Objektverarbeitung werden die zu prüfenden Objekte anhand ihres Status gefiltert. Diese Funktion basiert auf den folgenden Konfigurationsoptionen:

- action_av
- action_av_infected
- action_av_notscanned
- action_av_deleted

Ausführliche Informationen zu diesen Optionen finden Sie in der Manpage *scep.cfg(5)*.

Für jedes verarbeitete Objekt wird zunächst die in der Option *action_av* konfigurierte Aktion ausgeführt. Wenn diese beispielsweise den Wert *'accept'* hat, wird das Objekt akzeptiert. Bei den Werten *'defer'*, *'discard'* bzw. *'reject'* wird das Objekt dementsprechend temporär abgewiesen, verworfen bzw. endgültig abgewiesen. Ist die Option *scan* eingestellt, so wird das Objekt auf Viren geprüft. Wenn zusätzlich die Option *av_clean_mode* auf *yes* gesetzt ist, wird es außerdem gesäubert. Je nach dem Ergebnis der Prüfung lässt sich die weitere Verarbeitung dann mit den Konfigurationsoptionen *action_av_infected*, *action_av_notscanned* und *action_av_deleted* steuern. Ergibt sich anhand einer dieser drei Optionen die Aktion *'accept'*, so wird das Objekt akzeptiert, anderenfalls blockiert.

Benutzerspezifische Konfigurationen

Benutzerspezifische Konfigurationen ermöglichen eine noch flexiblere, individuellere Anpassung der Systemfunktionen. Der Administrator kann damit die Einstellungen für den SCEP-Virenschutz davon abhängig machen, welcher Benutzer auf die Dateisystem-Objekte zugreift.

Eine ausführliche Beschreibung dieser Funktion finden Sie in der Manpage *scep.cfg(5)*. Der vorliegende Abschnitt enthält nur ein kurzes Beispiel einer solchen benutzerspezifischen Konfiguration.

In diesem Beispiel soll das Modul *scep_dac* die Zugriffsarten ON_OPEN und ON_EXEC für einen externen Datenträger überwachen, der unter dem Verzeichnis /home eingehängt wurde. Das Modul kann im Abschnitt **[fac]** der SCEP-Konfigurationsdatei konfiguriert werden. Siehe hierzu das nachfolgende Beispiel:

```
[fac]
agent_type = "dazuko"
event_mask = "open"
ctl_incl = "/home"
action_av = "scan"
```

Um Prüfeinstellungen für einen bestimmten Benutzer zu definieren, muss im Parameter *'user_config'* der Name der speziellen Konfigurationsdatei angegeben werden, in der die individuellen Prüfregeln gespeichert sind. Im hier gezeigten Beispiel heißt diese Konfigurationsdatei *'scep_dac_spec.cfg'* und liegt im SCEP-Konfigurationsverzeichnis. (Der Pfad zu diesem Verzeichnis hängt vom Betriebssystem ab. Nähere Informationen hierzu finden Sie auf der Seite Begriffe und Abkürzungen.)

```
[fac]
agent_type = "dazuko"
event_mask = "open"
ctl_incl = "/home"
action_av = "scan"
user_config = "scep_dac_spec.cfg"
```

Nachdem Sie den Parameter *'user_config'* im Abschnitt **[fac]** festgelegt haben, müssen Sie dementsprechend die Datei *'scep_dac_spec.cfg'* im Konfigurationsverzeichnis anlegen und ihr die gewünschten Prüfregeln hinzufügen.

```
[username]
action_av = "reject"
```

Am Anfang des speziellen Konfigurationsabschnitts geben Sie den Benutzernamen an, für den die individuellen Prüfregeln gelten sollen. Im hier gezeigten Beispiel werden Dateisystem-Zugriffe aller anderen Benutzer normal verarbeitet, die Dateisystem-Objekte werden dabei also auf Infiltrationen geprüft. Zugriffe durch den Benutzer *username* werden hingegen verweigert (blockiert).

Taskplaner

Mit dem Taskplaner können Tasks automatisch zu einem festgelegten Zeitpunkt oder bei bestimmten Ereignissen ausgeführt werden. Zudem bietet er Funktionen zum Verwalten und Starten von Tasks mit vordefinierten Konfigurationen und Eigenschaften. Über die Konfiguration und die Eigenschaften eines Tasks kann nicht nur der Ausführungszeitpunkt festgelegt, sondern auch ein benutzerdefiniertes Profil für die Ausführung verwendet werden, was die Flexibilität weiter steigert.

Die Option *'scheduler_tasks'* ist standardmäßig auskommentiert, sodass die Standardkonfiguration des Taskplaners verwendet wird. In der SCEP-Konfigurationsdatei sind alle Parameter und Tasks durch Strichpunkte voneinander getrennt. Soll in einem Wert ein Strichpunkt oder Backslash verwendet werden, muss dieser daher durch einen vorangestellten Backslash codiert werden. Jeder Task hat sechs Parameter, deren Syntax wie folgt lautet:

- id - Eindeutige Nummer.
- name - Kurze Beschreibung des Tasks.
- flags - Hier können spezielle Flags zum Deaktivieren eines Tasks gesetzt werden.
- failstart - Legt fest, was geschehen soll, wenn der Task nicht zum festgelegten Zeitpunkt ausgeführt werden konnte.
- datespec - Standardkonforme Datumsangabe mit sechs Feldern (gleiches Format wie in crontab inkl. Jahr), Zeitintervall oder Ereignisoption.
- command - Entweder der absolute Pfad zu einem Befehl gefolgt von dessen Argumenten oder aber der Name eines Spezialbefehls mit dem Präfix „@“ (Beispiel: Virenschutz-Update = *@update*).

```
#scheduler_tasks = "id;name;flags;failstart;datespec;command;id2;name2;...";
```

Beim Parameter „datespec“ können anstelle einer Datumsangabe die folgenden Ereignisoptionen verwendet werden:

- start - Ausführung beim Daemon-Start.
- startonce - Ausführung beim Daemon-Start, höchstens jedoch einmal pro Tag.
- engine - Ausführung nach erfolgreichem Update des Prüfmoduls.
- login - Ausführung bei Anmeldung in Web-Oberfläche.
- threat - Ausführung, wenn eine Bedrohung erkannt wurde.
- notscanned - Nicht geprüfte Datei

Die aktuelle Taskplaner-Konfiguration können Sie sich über die Web-Oberfläche oder mit dem folgenden Befehl anzeigen lassen:

```
cat @ETCDIR@/scep.cfg | grep scheduler_tasks
```

Eine ausführliche Beschreibung des Taskplaners und seiner Parameter finden Sie im entsprechenden Abschnitt der Manpage *scep_daemon(8)*.

Web-Oberfläche

Die Web-Oberfläche ermöglicht eine komfortable Konfiguration und Administration von SCEP. Sie ist als separater Agent implementiert und muss daher gesondert aktiviert werden. Zur schnellen Konfiguration der *Web-Oberfläche* richten Sie die folgenden Optionen in der SCEP-Konfigurationsdatei ein und starten den SCEP-Daemon anschließend neu:

```
[wwwi]
agent_enabled = yes
listen_addr = address
listen_port = port
username = name
password = pass
```

Konfigurieren Sie dabei die Parameterwerte entsprechend Ihrer Umgebung. Anschließend können Sie die Web-Oberfläche in Ihrem Browser über die Adresse *'https://address:port'* aufrufen (die Platzhalter stehen für die Werte, die Sie oben festgelegt haben; achten Sie bitte auch auf das 'https'). Zur Anmeldung verwenden Sie die Angaben aus den Konfigurationsparametern *username/password*. Einführende Hinweise zur Verwendung finden Sie auf der Hilfeseite, technische Details zu *scep_wwwi* in der Manpage *scep_wwwi(1)*.

Über die Web-Oberfläche können Sie remote auf den SCEP-Daemon zugreifen und ihn ohne großen Aufwand bereitstellen. Außerdem erleichtert sie Ihnen das Lesen und Schreiben von Konfigurationswerten.

Abb. 6-1: System Center Endpoint Protection - Startseite

Die Web-Oberfläche von System Center Endpoint Protection ist in zwei Bereiche unterteilt. Im primären Fensterbereich sehen Sie den Inhalt der ausgewählten Menüoption sowie das Hauptmenü. Über die Leiste oben quer am Bildschirm können Sie auf die folgenden Optionen zugreifen:

- **Startseite** - Grundlegende Informationen zum System und zu Ihrem Microsoft-Produkt
- **Konfiguration** - Hier können Sie die Systemkonfiguration von System Center Endpoint Protection ändern.
- **Steuerung** - Hier können Sie eine Reihe von Verwaltungsaufgaben durchführen und globale Statistiken zu Objekten einsehen, die mit scep_daemon verarbeitet wurden.
- **Hilfe** - Ausführliche Anleitung, wie Sie die Web-Oberfläche von System Center Endpoint Protection verwenden
- **Abmelden** - Aktuelle Sitzung beenden

Wichtig: Denken Sie daran, Änderungen im Bereich **Konfiguration** der Web-Oberfläche immer mit einem Klick auf **Änderungen speichern** zu speichern, da sie sonst verlorengehen. Damit die Änderungen wirksam werden, müssen Sie außerdem noch den SCEP-Daemon neu starten, indem Sie im linken Fensterbereich auf **Änderungen übernehmen** klicken.

Beispiele für Konfiguration des Echtzeit-Schutzes

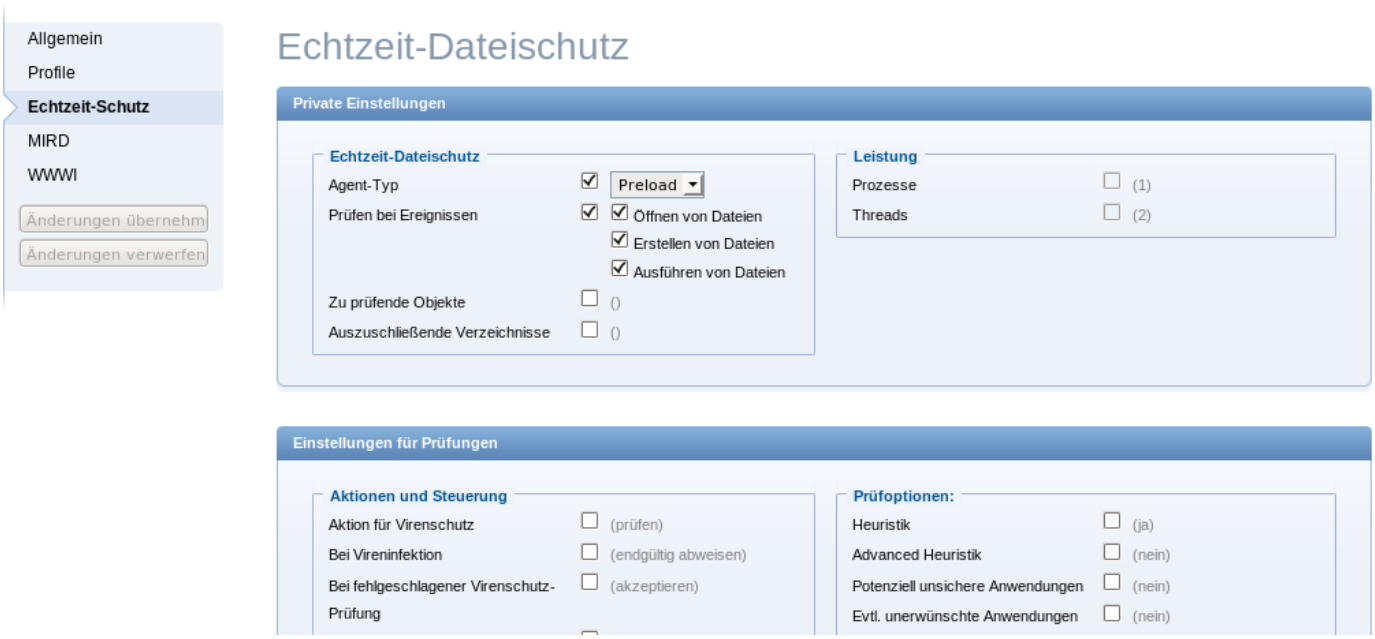
SCEP kann auf zwei Weisen konfiguriert werden. In diesem Beispiel zeigen wir Ihnen am Beispiel des Zugriffssteuerungsmoduls (siehe Kapitel Echtzeit-Schutz über Preload-Bibliothek für libc), wie Sie diese beiden Methoden verwenden. Sie können sich dann für die Methode entscheiden, die für Sie am praktischsten ist.

- Konfiguration über die SCEP-Konfigurationsdatei:

```
[fac]
agent_type = "preload"
event_mask = "open"
ctl_incl = "/home"
action_av_deleted = "reject"
action_av = "scan"
action_av_infected = "reject"
```

- Konfiguration über die Web-Oberfläche:

Abb. 6-3: SCEP - Konfiguration > Echtzeit-Schutz



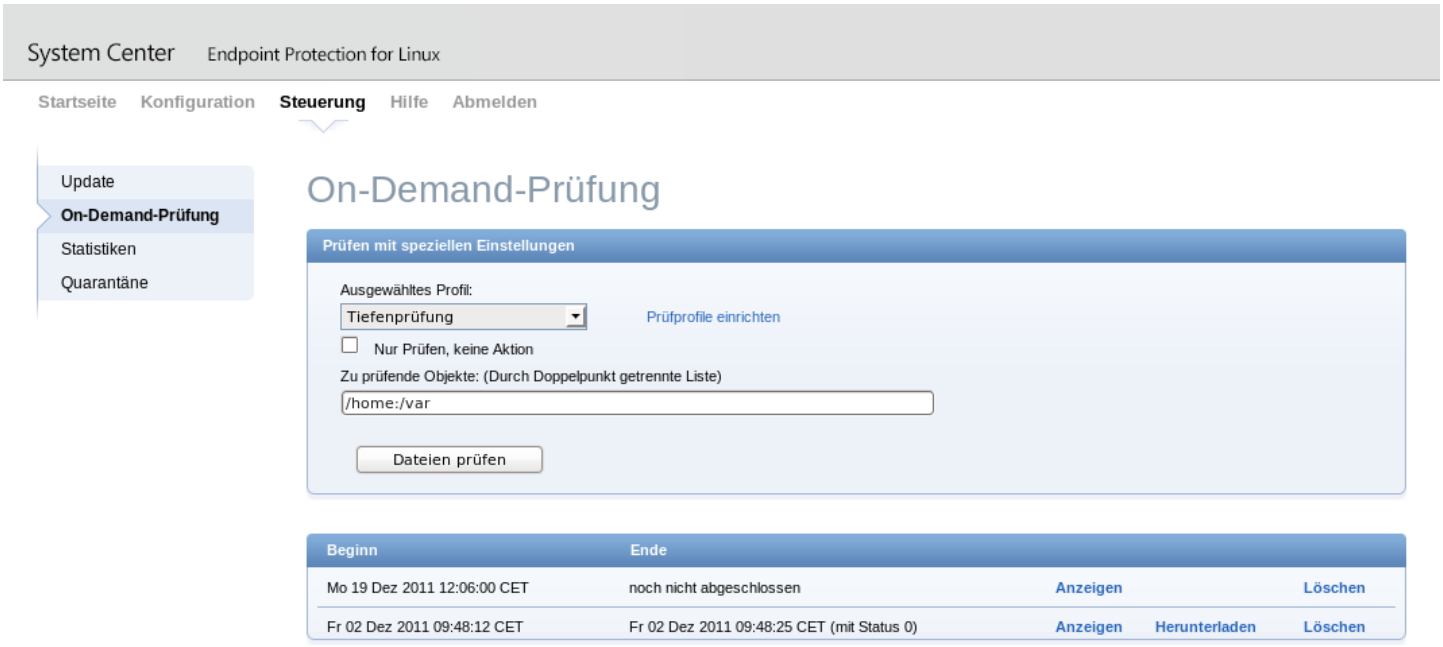
Wenn Sie Einstellungen über die Web-Oberfläche ändern, denken Sie bitte immer daran, sie mit einem Klick auf **Änderungen speichern** zu speichern. Anschließend müssen Sie die Änderungen noch übernehmen, indem Sie im Bereich **Konfiguration** auf **Änderungen übernehmen** klicken.

On-Demand-Prüfung

Dieser Abschnitt zeigt an einem Beispiel, wie Sie eine On-Demand-Prüfung zur Malware-Suche starten:

- Rufen Sie den folgenden Menüpunkt auf: **Steuerung > On-Demand-Prüfung**
- Geben Sie den Pfad zu dem Verzeichnis an, das geprüft werden soll.
- Klicken Sie auf **Dateien prüfen**, um den Kommandozeilen-Scanner zu starten.

Abb. 6-4: SCEP - Steuerung > On-Demand-Prüfung

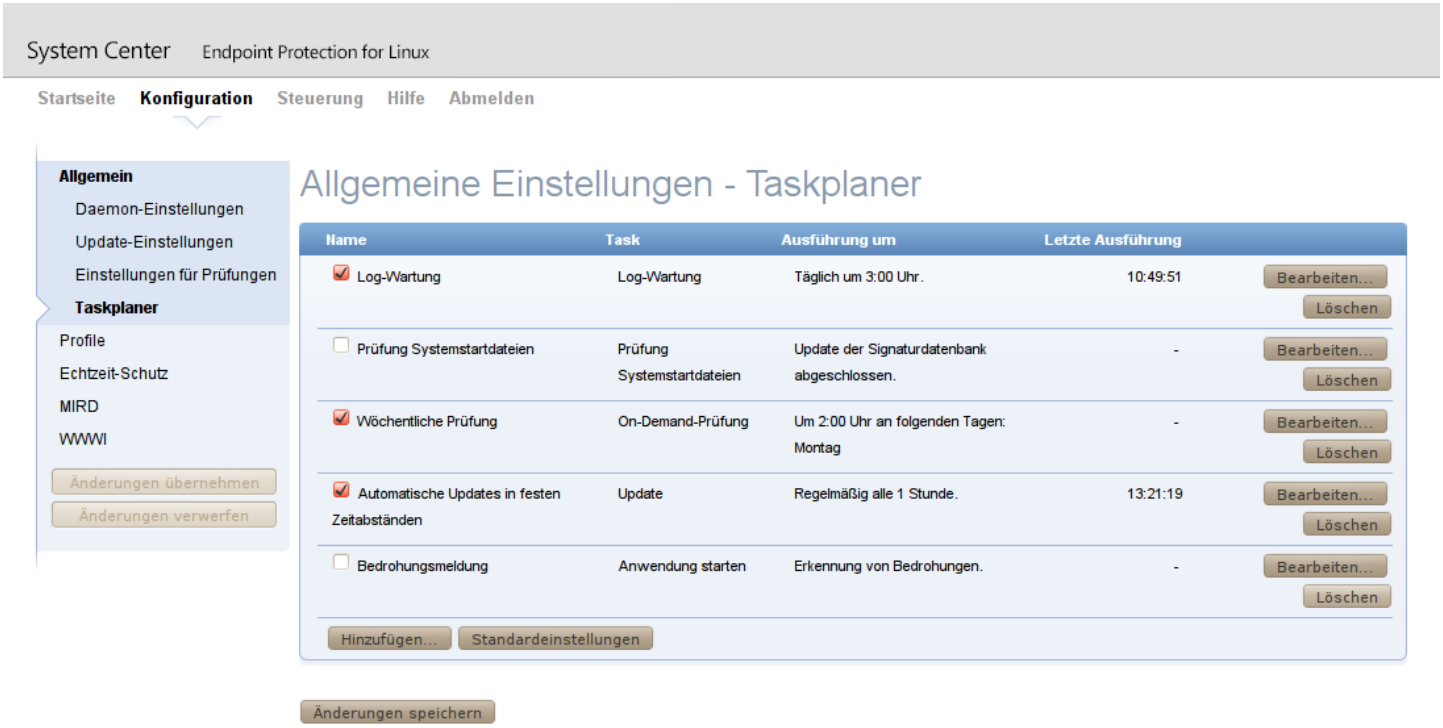


Der Microsoft-Kommandozeilen-Scanner wird automatisch im Hintergrund ausgeführt. Informationen zum Status der Prüfung können Sie über den Link **Anzeigen** aufrufen. Hiermit wird ein neues Browserfenster geöffnet.

Taskplaner

Die Tasks des Taskplaners können Sie entweder über die SCEP-Konfigurationsdatei (siehe Kapitel Taskplaner) oder über die Web-Oberfläche verwalten.

Abb. 6-5: SCEP - Allgemein > Taskplaner



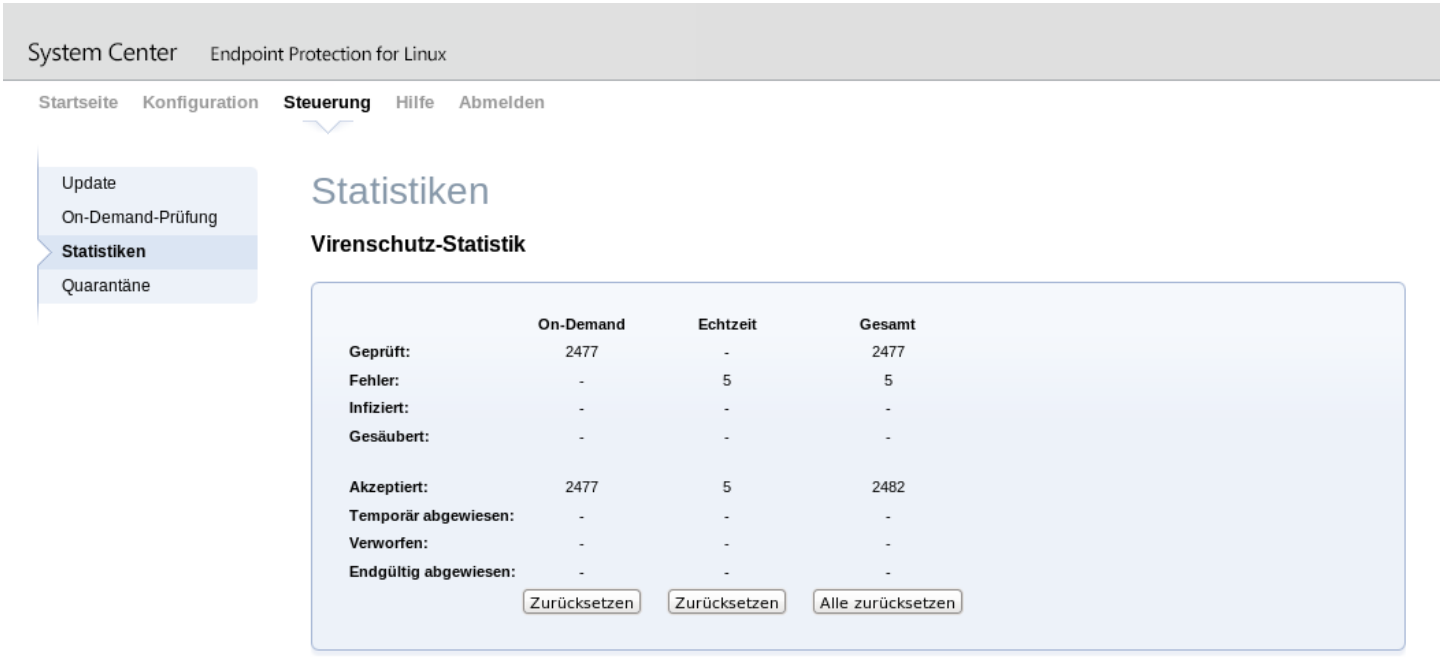
Um einen geplanten Task zu aktivieren oder zu deaktivieren, verwenden Sie das entsprechende Kontrollkästchen. Standardmäßig werden die folgenden Tasks angezeigt:

- **Log-Wartung** - Um den Speicherbedarf zu reduzieren, werden ältere Logs automatisch gelöscht. Außerdem wird eine Defragmentierung der Logs durchgeführt. Hierbei werden alle leeren Log-Einträge entfernt, was die Geschwindigkeit bei der Arbeit mit den Logs erhöht. Eine starke Verbesserung ist insbesondere dann erkennbar, wenn die Logs eine große Anzahl an Einträgen enthalten.
- **Prüfung Systemstartdateien** - Bewirkt, dass nach einem erfolgreichen Signaturdatenbank-Update der Arbeitsspeicher und die ausgeführten Dienste erneut geprüft werden.
- **Wöchentliche Prüfung** - Wöchentliche Prüfung des gesamten Dateisystems (standardmäßig jeden Montag um 2 Uhr nachts). Dieser Task kann vom Benutzer angepasst werden.
- **Automatische Updates in festen Zeitabständen** - Den optimalen Schutz Ihres Computers gewährleisten Sie, indem Sie System Center Endpoint Protection regelmäßig aktualisieren. Weitere Informationen finden Sie im Abschnitt SCEP-Update-Dienstprogramm.
- **Bedrohungsmeldung** - Standardmäßig wird jede erkannte Bedrohung in syslog protokolliert. Zusätzlich kann SCEP mit diesem Task so konfiguriert werden, dass ein externes Skript (Notifikationsskript) ausgeführt wird, um einen Administrator per E-Mail über die erkannte Bedrohung zu informieren.

Statistiken

Hier können Sie sich Statistiken zu allen aktiven Agents anzeigen lassen. Die Übersichtsseite **Statistiken** wird alle 10 Sekunden aktualisiert.

Abb. 6-6: SCEP - Steuerung > Statistiken



Logging

SCEP unterstützt ein Logging über den System-Daemon syslog. *Syslog* ist ein Standard zum Protokollieren von Programmmeldungen, kann aber auch zur Aufzeichnung von Systemereignissen wie Netzwerk- oder Sicherheitsereignissen verwendet werden.

Jede Meldung ist mit einer sogenannten „Facility“ gekennzeichnet:

```
auth, authpriv, daemon, cron, ftp, lpr, kern, mail, ..., local0, ..., local7
```

Außerdem erhält jede Meldung von ihrem Absender eine Priorität:

```
Error, Warning, Summall, Summ, Partall, Part, Info, Debug
```

In diesem Abschnitt wird beschrieben, wie die Log-Ausgabe über syslog konfiguriert und ausgewertet wird. Die Option *'syslog_facility'* (Standardwert *'daemon'*) definiert die verwendete syslog-Facility. Die syslog-Einstellungen können Sie in der SCEP-Konfigurationsdatei oder über die Web-Oberfläche bearbeiten. Um die Logging-Klasse zu ändern, ändern Sie den Wert des Parameters *'syslog_class'*. In diese Einstellungen sollten Sie nur eingreifen, wenn Sie mit syslog vertraut sind. Nachstehend eine Beispielkonfiguration:

```
syslog_facility = "daemon"
syslog_class = "error:warning:summall"
```

Name und Speicherort der Log-Datei hängen von Ihrer syslog-Installation und -Konfiguration ab (z. B. rsyslog, syslog-ng usw.). Die Standard-Dateinamen für die syslog-Ausgabe lauten beispielsweise *'syslog'*, *'daemon.log'* usw. Um die syslog-Aktivität anzuzeigen, führen Sie in der Konsole einen der folgenden Befehle aus:

```
tail -f /var/log/syslog
tail -100 /var/log/syslog | less
cat /var/log/syslog | grep scep | less
```

Wichtig: Damit die Überwachung von SCEP für Linux durch System Center Operations Manager korrekt ausgeführt wird, muss sie erst in der SCEP-Konfigurationsdatei oder über die SCEP-Weboberfläche aktiviert werden. Stellen Sie sicher, dass der in dieser Datei enthaltene Parameter *'scom_enabled'* auf *'scom_enabled = yes'* eingestellt ist. Alternativ können Sie die entsprechende Einstellung in der Web-Oberfläche unter **Konfiguration** > **Global** > **Daemon-Einstellungen** > **SCOM aktiviert** vornehmen.

Aktualisieren von SCEP

SCEP-Update-Dienstprogramm

Um einen wirksamen Schutz durch System Center Endpoint Protection zu gewährleisten, muss die Signaturdatenbank ständig auf dem neuesten Stand gehalten werden. Hierzu wurde das Dienstprogramm *scep_update* entwickelt. Nähere Informationen finden Sie in der Manpage *scep_update(8)*. Falls die Internetverbindung des Servers über einen HTTP-Proxyserver läuft, müssen außerdem die Optionen *'proxy_addr'* und *'proxy_port'* definiert werden. Schließlich sind im gleichen Abschnitt auch noch die Optionen *'proxy_username'* und *'proxy_password'* anzugeben, wenn für den Zugriff auf den HTTP-Proxyserver ein Benutzername und ein Passwort erforderlich sind. Um ein Update zu starten, geben Sie den folgenden Befehl ein:

@SBINDIR@/scep_update

Um höchstmögliche Sicherheit für den Anwender zu gewährleisten, sammelt das Microsoft-Team fortlaufend Malware-Definitionen aus der ganzen Welt. Neue Signaturen werden in sehr kurzen Zeitabständen in die Signaturdatenbank aufgenommen. Daher empfiehlt es sich dringend, regelmäßige Updates einzurichten. Dies erreichen Sie, indem Sie den Task *'@update'* unter der Option *'scheduler_tasks'* im Abschnitt **[global]** der SCEP-Konfigurationsdatei konfigurieren. Alternativ können Sie das Update-Intervall über den Taskplaner festlegen. Damit das Signaturdatenbank-Update erfolgreich abgeschlossen werden kann, muss der SCEP-Daemon laufen.

Ablauf eines SCEP-Updates

Updates erfolgen in zwei Schritten. Zunächst werden die vorkompilierten Update-Module vom Microsoft-Server heruntergeladen.

Im zweiten Schritt werden anhand der Module, die im lokalen Update-Mirror gespeichert sind, die Module kompiliert, die vom System Center Endpoint Protection-Scanner geladen werden können. In der Regel werden die folgenden SCEP-Lademodule erstellt: Loader (em000.dat), Scanner (em001.dat), Signaturdatenbank (em002.dat), Archivunterstützung (em003.dat), Advanced Heuristik (em004.dat) usw. Die Module werden im folgenden Verzeichnis erstellt:

@BASEDIR@

Ihr Feedback an uns

Wir hoffen, dass Sie in diesem Handbuch alle gesuchten Informationen zur Installation, Konfiguration und Verwaltung von System Center Endpoint Protection gefunden haben. Natürlich möchten wir unsere Dokumentation fortlaufend verbessern, um sie so hilfreich wie möglich zu gestalten. Wenn Sie also Verbesserungswünsche zu diesem Handbuch haben, etwa weil bestimmte Abschnitte unklar oder unvollständig sind, teilen Sie dies bitte unserem Support mit:

support.microsoft.com

Auch bei anderen Problemen mit diesem Produkt hilft Ihnen unser Support gerne weiter.

Anhang A: PHP-Lizenz

Redistribution and use in source and binary forms, with or without modification, is permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. The name "PHP" must not be used to endorse or promote products derived from this software without prior written permission. For written permission, please contact group@php.net.
4. Products derived from this software may not be called "PHP", nor may "PHP" appear in their name, without prior written permission from group@php.net. You may indicate that your software works in conjunction with PHP by saying "Foo for PHP" instead of calling it "PHP Foo" or "phpfoo"
5. The PHP Group may publish revised and/or new versions of the license from time to time. Each version will be given a distinguishing version number. Once covered code has been published under a particular version of the license, you may always continue to use it under the terms of that version. You may also choose to use such covered code under the terms of any subsequent version of the license published by the PHP Group. No one other than the PHP Group has the right to modify the terms applicable to covered code created under this License.
6. Redistributions of any form whatsoever must retain the following acknowledgment: "This product includes PHP software, freely available from <http://www.php.net/software/>".

THIS SOFTWARE IS PROVIDED BY THE PHP DEVELOPMENT TEAM "AS IS" AND ANY EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE PHP DEVELOPMENT TEAM OR ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.