

Novell® Sentinel™

5.1.3

www.novell.com

Band IV - SENTINEL-REFERENZHANDBUCH

7. Juli 2006



Novell®

Rechtliche Hinweise

Novell, Inc., übernimmt keine Gewährleistung oder Haftung in Bezug auf den Inhalt und die Verwendung dieser Dokumentation und schließt insbesondere jede ausdrückliche oder stillschweigende Gewährleistung bezüglich der handelsüblichen Qualität sowie der Eignung für einen bestimmten Zweck aus.

Darüber hinaus behält sich Novell, Inc., das Recht vor, diese Veröffentlichung ohne vorherige Ankündigung zu überarbeiten und inhaltliche Änderungen vorzunehmen, ohne dass für Novell die Verpflichtung entsteht, Personen oder Organisationen über die vorgenommenen Änderungen zu informieren.

Novell, Inc., übernimmt ferner keine Gewährleistung oder Haftung in Bezug auf Software und schließt insbesondere jede ausdrückliche oder stillschweigende Gewährleistung bezüglich der handelsüblichen Qualität sowie der Eignung für einen bestimmten Zweck aus. Darüber hinaus behält sich Novell, Inc., das Recht vor, die Novell-Software vollständig oder teilweise zu ändern, ohne dass für Novell die Verpflichtung entsteht, Personen oder Organisationen über die vorgenommenen Änderungen zu informieren.

Sämtliche Produkte und technischen Informationen, die im Rahmen dieser Vereinbarung bereitgestellt werden, unterliegen möglicherweise den US-Exportbestimmungen und den Handelsgesetzen anderer Länder. Hiermit erklären Sie sich bereit, sämtliche Exportbestimmungen einzuhalten und ggf. die erforderlichen Lizenzen oder Berechtigungen für den Export, die Wiederausfuhr oder den Import einzuholen. Sie erklären sich bereit, keinen Export oder keine Wiederausfuhr an natürliche oder juristische Personen zu tätigen, die zurzeit auf den Exportausschlusslisten der USA aufgeführt sind, oder in Länder, die einem Embargo unterliegen oder die den US-Exportbestimmungen zufolge den Terrorismus unterstützen. Sie erklären sich bereit, die Lieferbestandteile nicht für die Endnutzung in verbotenen nuklearen, chemischen oder biologischen Waffen oder Raketen einzusetzen.

Weitere Informationen zum Export von Novell-Software finden Sie unter www.novell.com/info/exports/. Novell übernimmt keinerlei Verantwortung, wenn Sie es versäumen, die erforderlichen Exportgenehmigungen einzuholen.

Copyright © 1999–2006, Novell, Inc. Alle Rechte vorbehalten. Ohne ausdrückliche, schriftliche Genehmigung des Herausgebers darf kein Teil dieser Veröffentlichung reproduziert, fotokopiert, übertragen oder in einem Speichersystem verarbeitet werden.

Novell, Inc., besitzt Rechte an geistigem Eigentum für die Technologie, die in das in dieser Dokumentation beschriebene Produkt integriert ist. Diese Rechte an geistigem Eigentum umfassen im Besonderen eines oder mehrere der unter <http://www.novell.com/company/legal/patents/> aufgelisteten Patente sowie ein oder mehrere andere Patente oder Patentanmeldungen in den USA und in anderen Ländern, sind jedoch nicht darauf beschränkt.

Novell, Inc.
404 Wyman Street, Suite 500
Waltham, MA 02451
U.S.A.
www.novell.com

Online-Dokumentation: Zugriff auf die Online-Dokumentation für dieses und andere Novell-Produkte sowie auf Aktualisierungen erhalten Sie unter www.novell.com/documentation.

Novell-Marken

Informationen zu Novell-Marken finden Sie in der Liste der Marken und Dienstleistungsmarken von Novell (<http://www.novell.com/company/legal/trademarks/tmlist.html>).

Materialien von Drittanbietern

Alle Marken von Drittanbietern sind Eigentum der jeweiligen Inhaber.

Rechtliche Hinweise zu Drittanbieterprodukten

Sentinel 5 enthält möglicherweise folgende Drittanbietertechnologien:

- Apache Axis und Apache Tomcat, Copyright © 1999 bis 2005, Apache Software Foundation. Weitere Informationen, Haftungsausschlüsse und Beschränkungen finden Sie unter <http://www.apache.org/licenses/>
- ANTLR. Weitere Informationen, Haftungsausschlüsse und Beschränkungen finden Sie unter <http://www.antlr.org>
- Boost, Copyright © 1999, Boost.org.
- Bouncy Castle, Copyright © 2000–2004, the Legion of Bouncy Castle. Weitere Informationen, Haftungsausschlüsse und Beschränkungen finden Sie unter <http://www.bouncycastle.org>.
- Checkpoint. Copyright © Check Point Software Technologies Ltd.
- Concurrent, Dienstprogrammpaket. Copyright © Doug Lea. Wird ohne die Klassen CopyOnWriteArrayList und ConcurrentReaderHashMap verwendet.
- Crypto++ Compilation. Copyright © 1995–2003, Wei Dai, beinhaltet folgende durch Copyright geschützte Werke: mars.cpp von Brian Gladman und Sean Woods. Weitere Informationen, Haftungsausschlüsse und Beschränkungen finden Sie unter <http://www.eskimo.com/~weidai/License.txt>.
- Crystal Reports Developer und Crystal Reports Server. Copyright © 2004 Business Objects Software Limited.
- DataDirect Technologies Corp. Copyright © 1991–2003.
- edpFTPj, lizenziert unter: Lesser GNU Public License. Weitere Informationen, Haftungsausschlüsse und Beschränkungen finden Sie unter <http://www.enterprisedt.com/products/edtfpj/purchase.html>.
- Enhydra Shark, lizenziert unter der Lesser General Public License, verfügbar unter: <http://shark.objectweb.org/license.html>.
- ICEsoft ICEbrowser. ICEsoft Technologies, Inc. Copyright © 2003–2004.
- ILOG, Inc. Copyright © 1999–2004.
- Installshield Universal. Copyright © 1996–2005, Macrovision Corporation und/oder Macrovision Europe Ltd.
- Java 2 Platform, Standard Edition. Copyright © Sun Microsystems, Inc. Weitere Informationen, Haftungsausschlüsse und Beschränkungen finden Sie unter http://java.sun.com/j2se/1.4.2/j2re-1_4_2_10-license.txt.

Java 2 Platform kann außerdem folgende Drittanbieterprodukte enthalten:

- CoolServlets © 1999
- DES and 3xDES © 2000, Jef Poskanzer
- Crimson © 1999–2000, The Apache Software Foundation
- Xalan J2 © 1999–2000, The Apache Software Foundation
- NSIS 1.0j © 1999–2000, Nullsoft, Inc.

- Eastman Kodak Company © 1992
- Lucinda, eine eingetragene Marke oder Marke von Bigelow and Holmes
- Taligent, Inc.
- IBM, einige Teile verfügbar unter: <http://oss.software.ibm.com/icu4j/>

Weitere Informationen zu diesen Drittanbietertechnologien und den zugehörigen Haftungsausschlüssen und Einschränkungen finden Sie unter: http://java.sun.com/j2se/1.4.2/j2se-1_4_2-thirdpartylicensereadme.txt.

- JavaBeans Activation Framework (JAF). Copyright © Sun Microsystems, Inc. Weitere Informationen, Haftungsausschlüsse und Beschränkungen finden Sie unter <http://www.java.sun.com/products/javabeans/glasgow/jaf.html>, klicken Sie auf "Download" > "License".
- JavaMail. Copyright © Sun Microsystems, Inc. Weitere Informationen, Haftungsausschlüsse und Beschränkungen finden Sie unter <http://www.java.sun.com/products/javamail/downloads/index.html>, klicken Sie auf „Download“ > „License“.
- Java Ace von Douglas C. Schmidt und seiner Forschungsgruppe an der Washington University und Tao (mit ACE-Wrappers) von Douglas C. Schmidt und seiner Forschungsgruppe an der Washington University, University of California, Irvine, und Vanderbilt University. Copyright © 1993–2005. Weitere Informationen, Haftungsausschlüsse und Beschränkungen finden Sie unter <http://www.cs.wustl.edu/~schmidt/ACE-copying.html> und <http://www.cs.wustl.edu/~pjain/java/ace/JACE-copying.html>
- Java Authentication and Authorization Service Modules, lizenziert unter: Lesser General Public License. Weitere Informationen, Haftungsausschlüsse und Beschränkungen finden Sie unter <http://free.tagish.net/jaas/index.jsp>.
- Java Network Launching Protocol (JNLP). Copyright © Sun Microsystems, Inc. Weitere Informationen, Haftungsausschlüsse und Beschränkungen finden Sie unter <http://www.java.sun.com/products/javawebstart/download-jnlp.html>, klicken Sie auf „Download“ > „License“.
- Java Service Wrapper. Teile wie folgt durch Copyright geschützt: Copyright © 1999, 2004 Tanuki Software und Copyright © 2001 Silver Egg Technology. Weitere Informationen, Haftungsausschlüsse und Beschränkungen finden Sie unter <http://wrapper.tanukisoftware.org/doc/english/license.html>.
- JIDE. Copyright © 2002 bis 2005, JIDE Software, Inc.
- jTDS ist lizenziert unter: Lesser GNU Public License. Weitere Informationen, Haftungsausschlüsse und Beschränkungen finden Sie unter <http://jtds.sourceforge.net/>.
- MDateSelector. Copyright © 2005, Martin Newstead, lizenziert unter: Lesser General Public License. Weitere Informationen, Haftungsausschlüsse und Beschränkungen finden Sie unter <http://web.ukonline.co.uk/mseries>
- Monarch Charts. Copyright © 2005, Singleton Labs.
- Net-SNMP. Teile des Codes unterliegen dem Copyright verschiedener juristischer Personen, die sich alle Rechte vorbehalten. Copyright © 1989, 1991, 1992 by Carnegie Mellon University; Copyright © 1996, 1998 bis 2000, the Regents of the University of California; Copyright © 2001 bis 2003 Networks Associates Technology, Inc.; Copyright © 2001 bis 2003, Cambridge Broadband, Ltd.; Copyright © 2003 Sun Microsystems, Inc., und Copyright © 2003 bis 2004, Sparta, Inc. Weitere Informationen, Haftungsausschlüsse und Beschränkungen finden Sie unter <http://net-snmp.sourceforge.net>.
- The OpenSSL Project. Copyright © 1998–2004, The Open SSL Project. Weitere Informationen, Haftungsausschlüsse und Beschränkungen finden Sie unter <http://www.openssl.org>.
- Oracle Help für Java. Copyright © 1994–2006, Oracle Corporation.
- RoboHELP Office. Copyright © Adobe Systems Incorporated, vormals Macromedia.
- Skin Look and Feel (SkinLF). Copyright © 2000–2006 L2FProd.com. Lizenziert unter der Apache-Softwarelizenz. Weitere Informationen, Haftungsausschlüsse und Beschränkungen finden Sie unter <https://skinlf.dev.java.net/>.
- Sonic Software Corporation. Copyright © 2003–2004. Die SSC-Software enthält Sicherheitssoftware, die von RSA Security, Inc., lizenziert wurde.

- Tinyxml. Weitere Informationen, Haftungsausschlüsse und Beschränkungen finden Sie unter <http://grinninglizard.com/tinyxmldocs/index.html>.
- SecurityNexus. Copyright © 2003 bis 2006. SecurityNexus, LLC. Alle Rechte vorbehalten.
- Xalan und Xerces, jeweils von der Apache Software Foundation lizenziert, Copyright © 1999–2004. Weitere Informationen, Haftungsausschlüsse und Beschränkungen finden Sie unter <http://xml.apache.org/dist/LICENSE.txt>.
- yWorks. Copyright © 2003 bis 2006, yWorks.

HINWEIS: Zum Zeitpunkt der Veröffentlichung dieser Dokumentation waren die oben stehenden Links aktiv. Sollten Sie feststellen, dass einer der oben angegebenen Links unterbrochen oder die verlinkten Webseiten inaktiv sind, wenden Sie sich an Novell, Inc., 404 Wyman Street, Suite 500, Waltham, MA 02451 U.S.A.

Vorwort

Bei der technischen Dokumentation von Sentinel handelt es sich um allgemeine, zweckorientierte Handbücher für den Betrieb und zur Referenz. Diese Dokumentation ist für Mitarbeiter des Bereichs Informationssicherheit konzipiert. Der Text in dieser Dokumentation gilt als Referenzquelle zum Enterprise Security Management System von Novell. Im Novell-Webportal steht weitere Dokumentation zur Verfügung.

Die Technische Dokumentation von Sentinel umfasst fünf einzelne Ausgaben. Dazu gehören:

- Band I – Sentinel™ 5-Installationshandbuch
- Band II – Sentinel™ 5-Benutzerhandbuch
- Band III – Sentinel™ 5 Wizard-Benutzerhandbuch
- Band IV – Sentinel™-Referenzhandbuch für Benutzer
- Band V – Sentinel™-Handbuch für Drittanbieter-Integration

Band I – Sentinel Installationshandbuch

In diesem Handbuch wird die Installation folgender Komponenten erläutert:

- Sentinel Server
- Sentinel Console
- Sentinel Correlation Engine
- Sentinel Crystal Reports
- Wizard Collector Builder
- Wizard Collector Manager
- Advisor

Band II – Sentinel Benutzerhandbuch

In diesem Handbuch werden die folgenden Themen behandelt:

- Verwendung der Sentinel Console
- Sentinel-Funktionen
- Sentinel Architektur
- Sentinel Kommunikation
- Herunterfahren/Starten von Sentinel
- Anfälligkeitsbewertung
- Ereignisüberwachung
- Ereignisfilterung
- Ereigniskorrelation
- Sentinel Data Manager
- Ereigniskonfiguration für Unternehmensrelevanz
- Zuordnungsservice
- Verlaufsberichte
- Wizard-Host-Verwaltung
- Vorfälle
- Szenarios
- Benutzerverwaltung
- Workflow

Band III – Wizard-Benutzerhandbuch

In diesem Handbuch werden die folgenden Themen behandelt:

- Wizard Collector Builder-Operation
- Wizard Collector Manager
- Collectors
- Wizard-Host-Verwaltung
- Erstellen und Verwalten von Collectors

Band IV – Sentinel Referenzhandbuch für Benutzer

In diesem Handbuch werden die folgenden Themen behandelt:

- Wizard-Skriptsprache
- Wizard-Parsing-Befehle
- Wizard-Administratorfunktionen
- META-Tags für Wizard und Sentinel
- Benutzerberechtigungen
- Sentinel Correlations Engine
- Correlations-Befehlszeilenoptionen
- Sentinel-Datenbankschema

Volume V – Sentinel Handbuch für Drittanbieter-Integration

- Remedy
- HP OpenView Operations
- HP Service Desk

Inhalt

1 Sentinel™ 5: Einführung zum Referenzhandbuch für Benutzer.....	1-1
Inhalt	1-1
Verwendete Konventionen.....	1-2
Hinweise und Warnhinweise	1-2
Befehle	1-2
Weitere Sentinel-Referenzen.....	1-2
Kontaktaufnahme mit Novell.....	1-2
2 Wizard-Skriptsprache.....	2-1
decide-Zeichenketten	2-1
Bearbeiten des Rx Buffer Pointer (Zeiger des Empfangspuffers)	2-1
Format	2-2
Parameternamen.....	2-2
Vorgangshierarchie in einer decide-Zeichenkette	2-2
Regeln für den Receive Buffer Pointer.....	2-3
Prüfen auf leeren Empfangspuffer	2-3
Beispiel für Auswertungen und Ergebnisse von decide-Zeichenketten	2-3
Reguläre Ausdrücke	2-4
Zusammenfassung von Sonderzeichen für reguläre Ausdrücke	2-4
Leerzeichen in regulären Ausdrücken.....	2-5
Parsing-Befehle	2-5
Einfache Datentypen	2-6
Derived Aggregate-Datentypen.....	2-7
Sonderregeln für Variablen	2-7
3 Wizard-Parsing-Befehle	3-1
Befehlsformat und Verwendung von Arrays	3-3
Befehle.....	3-4
ALERT	3-4
APPEND	3-5
BITFIELD.....	3-8
BREAKPOINT	3-9
BYTEFIELD	3-10
CLEAR.....	3-12
CLEARTAGS.....	3-13
COMMENT	3-14
COMPARE	3-14
CONSTANTTAGS	3-15
CONVERT	3-16
COPY	3-18
CRC	3-20
DATE	3-21
DATETIME	3-22
DBCLOSE	3-23

DBDELETE	3-23
DBGETROW	3-24
DBINSERT	3-25
DBOPEN	3-26
DBSELECT	3-27
DEC	3-28
DECODE	3-29
DECODEMIME	3-29
DELETE	3-30
DISPLAY	3-31
ELSE	3-32
ENCODE	3-33
ENCODEMIME	3-33
ENDFOR	3-34
ENDIF	3-34
ENDWHILE	3-35
EVENT	3-35
FILEA	3-38
FILEL	3-39
FILER	3-40
FILEW	3-41
FOR	3-42
GETCONFIG	3-43
GETENV	3-44
HEXTONUM	3-44
IF 3-45	
INC	3-47
INDICATOR	3-47
INFO_CLEAR_TAGS	3-48
INFO_CLOSE	3-48
INFO_CONSTANT_TAGS	3-49
INFO_CREATE	3-49
INFO_DUMP	3-50
INFO_PUSH	3-50
INFO_SEND	3-51
INFO_SETTAG	3-51
INFO_*-Befehlsbeispiel	3-54
IPTONUM	3-55
LENGTH oder LENGTH-OPTION2	3-56
LOOKUP	3-57
NEGSEARCH	3-59
NUMTOHEX	3-60
NUMTOIP	3-60
PARSER_ATTACHVARIABLE	3-61
PARSER_CREATEBASIC	3-63
PARSER_NEXT	3-64
PARSER_PARSESTRING	3-64
PAUSE	3-65
POPUP	3-65
PRINTF	3-66
REGEXP_REPLACE	3-68
REGEXPSEARCH, REGEXPSEARCH_EXPLICIT bzw. REGEXPSEARCH_STRING	3-70

REPLACE	3-73
RESET	3-74
RXBUFF	3-74
SEARCH	3-75
SET	3-76
SETBYTES	3-77
SETCONFIG	3-78
SHELL	3-79
SKIP	3-80
SKIPWORD	3-81
SOCKETW	3-83
STONUM	3-84
STRIP bzw. STRIP-ASCII-RANGE	3-84
TBOSETCOMMAND	3-85
TBOSETREQUEST	3-88
TIME	3-90
TOKENIZE	3-90
TOLOWER	3-92
TOUPPER	3-92
TRANSLATE	3-93
TRIM	3-95
WHILE	3-96
4 Wizard-Administratorfunktionen	4-1
Wizard-Dienstprogramme und -Anwendungen	4-1
Collector Builder	4-1
Collector Manager	4-1
Collector Engine	4-2
popup.exe	4-2
popup.cfg	4-2
Wizard-Verzeichnisstruktur	4-3
5 META-Tags für Wizard und Sentinel	5-1
6 Benutzerberechtigungen für Sentinel Control Center	6-1
Standardbenutzer	6-1
Allgemein	6-2
Allgemein – Öffentliche Filter	6-2
Allgemein – Private Filter	6-2
Allgemein – Integrationsaktionen	6-2
Active Views	6-3
Active Views – Menüelemente	6-3
Active Views – Zusammenfassungansichten	6-3
iTRAC	6-3
Schablonenverwaltung	6-3
Vorgangsverwaltung	6-4
Vorfälle	6-4
Collector-Verwaltung	6-4
Analyse	6-5
Advisor	6-5
Verwaltung	6-5

Verwaltung – Korrelation	6-6
Verwaltung – Globale Filter	6-6
Verwaltung – Menükonfiguration	6-6
Verwaltung – DAS-Statistik	6-6
Verwaltung – Ereignisdatei-Info	6-6
Verwaltung – Serveransichten	6-6
Verwaltung – Benutzerverwaltung	6-6
Verwaltung – Benutzersitzungsverwaltung	6-7
Verwaltung – iTRAC-Funktionsverwaltung	6-7
7 Sentinel Correlation Engine.....	7-1
Korrelationsfiltertypen	7-2
Mustertypen-Korrelationsfilter	7-2
Filtermanager-Typ-Korrelationsfilter	7-3
Editortyp-Korrelationsfilter	7-3
Definition von Korrelationsregeln.....	7-5
Beobachtungsliste	7-5
Grundlegende Korrelation	7-5
Erweiterte Korrelation	7-6
FreeForm RuleLg-Korrelation.....	7-6
Erstellen einer Regel für die Beobachtungsliste.....	7-6
Erstellen einer grundlegenden Korrelationsregel	7-10
Erstellen einer erweiterten Korrelationsregel.....	7-14
Erstellen einer FreeForm RuleLg-Korrelationsregel	7-19
Filteroperation.....	7-20
Fensteroperation („window“)	7-21
Auslöseoperation („trigger“).....	7-23
Operatoren, die gemeinsam mit Operationen Regeln bilden	7-24
Beispiel für Korrelationsregeln.....	7-26
Pufferüberlauf-Angriff und Serviceunterbrechung.....	7-27
Denial of Service-Angriff und Serviceunterbrechung	7-27
Ermitteln eines Virenausbruchs.....	7-28
Ermitteln eines Wurmausbruchs	7-28
Ermitteln von Trojanern	7-29
Multiple Backdoor-Angriffsversuche von einer einzelnen Quelle	7-29
Multiple Backdoor-Angriffsversuche von verschiedenen Quellen	7-30
Mehrere Fehler bei der Anmeldung von einer beliebigen Quelle bei einem beliebigen Ziel..	7-30
Mehrere Fehler bei der Anmeldung von derselben Quelle bei demselben Ziel.....	7-31
Pufferüberlauf-Angriff von derselben Quelle zu demselben Ziel.....	7-31
Brute Force-Angriff, bei dem Quelle und Ziel identisch sind	7-32
Microsoft – Überprüfung von IIS-Angriffen (Internet Information Services)	7-32
MDAC-Angriff (Microsoft – Microsoft Data Access Connector) – Überprüfung auf Remote	
Data Services-Angriff	7-33
Microsoft– SQL Server-Angriffe – Überprüfung auf SQL Server-Angriffe.....	7-33
Microsoft – NETBIOS – Überprüfung auf Angriffe auf ungesicherte Windows-	
Netzwerkfreigaben	7-34
Microsoft – Anonyme Anmeldung – Überprüfung auf Null-Sitzungs-Angriffe	7-34
Microsoft – LM-Authentifizierung (LAN Manager) – Überprüfung von Hashing-Angriffen auf	
schwachen LM	7-35
Microsoft – Überprüfung auf Angriffe auf allgemeine Windows-Authentifizierung	7-35
Microsoft – Überprüfung auf Internet Explorer-Angriffe (IE).....	7-35

Microsoft – Überprüfung auf Angriff für Remote-Zugriff auf Registrierung	7-36
Microsoft – Überprüfung auf Angriffe auf Windows Scripting	7-36
UNIX - Überprüfung auf RPC-Angriff (Remote Procedure Call)	7-37
UNIX – Überprüfung auf Angriffe auf Apache Web Server	7-37
UNIX – Überprüfung auf SSH-Angriff (Secure Shell)	7-38
UNIX – Überprüfung auf SNMP-Angriff (Simple Network Management Protocol)	7-38
UNIX – Überprüfung auf FTP-Angriffe (File Transfer Protocol)	7-39
UNIX - Überprüfung auf Remote-Services-Angriff	7-39
UNIX – Überprüfung auf Line Printer Daemon-Angriff	7-40
UNIX – Überprüfung auf Sendmail-Angriff	7-40
UNIX – Überprüfung auf BIND/DNS-Angriff	7-40
UNIX – Angriff auf allgemeine UNIX-Authentifizierung überprüft	7-41
Taxonomietabellen	7-41
NIDS-Taxonomietabelle	7-41
HIDS- und OS-Taxonomietabelle	7-45
Korrelationsausgabe	7-49
Korrelationsregel-Ausgabestruktur	7-49
Übergebene Skriptparameter	7-49
8 Sentinel-Korrelations-Befehlszeilenoptionen	8-1
9 Sentinel Data Access Service	9-1
DAS-Containerdateien	9-1
Erneutes Konfigurieren der Datenbankverbindungseigenschaften	9-2
DAS-Konfigurationsdateien	9-3
Native Datenbank-Connectors zum Einfügen von Ereignissen	9-4
10 Standardbenutzerpasswörter ändern	10-1
Standardbenutzerpasswörter für Oracle- und MS SQL-Authentifizierung ändern	10-1
Ändern des esecadm-Passworts	10-1
Ändern des esecapp-Passworts	10-1
Ändern des esecdba-Passworts	10-2
Ändern des esecrpt-Passworts	10-2
Standardbenutzerpasswörter für Windows-Authentifizierung ändern	10-3
Ändern des Sentinel-Administrator-Passworts	10-3
Ändern des Sentinel-DB-Administrator-Passworts	10-3
Ändern des Administrator-Passworts für Sentinel-Anwendungs-DB	10-4
Ändern des Sentinel Report-Benutzer-Passworts	10-5
11 Sentinel-Datenbankansichten für Oracle	11-1
Ansichten	11-1
ADV_ALERT_CVE_RPT_V	11-1
ADV_ALERT_PRODUCT_RPT_V	11-1
ADV_ALERT_RPT_V	11-2
ADV_ATTACK_ALERT_RPT_V	11-2
ADV_ATTACK_CVE_RPT_V	11-3
ADV_ATTACK_MAP_RPT_V	11-3
ADV_ATTACK_PLUGIN_RPT_V	11-3
ADV_ATTACK_RPT_V	11-4
ADV_CREDIBILITY_RPT_V	11-4

ADV_FEED_RPT_V	11-5
ADV_PRODUCT_RPT_V	11-5
ADV_PRODUCT_SERVICE_PACK_RPT_V	11-5
ADV_PRODUCT_VERSION_RPT_V	11-6
ADV_SEVERITY_RPT_V	11-6
ADV_SUBALERT_RPT_V	11-6
ADV_URGENCY_RPT_V	11-7
ADV_VENDOR_RPT_V	11-7
ADV_VULN_PRODUCT_RPT_V	11-8
ANNOTATIONS_RPT_V	11-8
ASSET_CTGRY_RPT_V	11-8
ASSET_HOSTNAME_RPT_V	11-9
ASSET_IP_RPT_V	11-9
ASSET_LOCATION_RPT_V	11-9
ASSET_RPT_V	11-10
ASSET_VALUE_RPT_V	11-10
ASSET_X_ENTITY_X_ROLE_RPT_V	11-10
ASSOCIATIONS_RPT_V	11-11
ATTACHMENTS_RPT_V	11-11
CONFIGS_RPT_V	11-11
CONTACTS_RPT_V	11-12
CORRELATED_EVENTS_RPT_V	11-12
CORRELATED_EVENTS_RPT_V1	11-12
CRITICALITY_RPT_V	11-13
CUST_RPT_V	11-13
ENTITY_TYPE_RPT_V	11-13
ENV_IDENTITY_RPT_V	11-14
ESEC_DISPLAY_RPT_V	11-14
ESEC_PORT_REFERENCE_RPT_V	11-15
ESEC_PROTOCOL_REFERENCE_RPT_V	11-15
ESEC_SEQUENCE_RPT_V	11-16
EVENTS_ALL_RPT_V (aus Gründen der Abwärtskompatibilität angegeben)	11-16
EVENTS_ALL_RPT_V1 (aus Gründen der Abwärtskompatibilität angegeben)	11-21
EVENTS_RPT_V (aus Gründen der Abwärtskompatibilität angegeben)	11-21
EVENTS_RPT_V1 (aus Gründen der Abwärtskompatibilität angegeben)	11-21
EVENTS_RPT_V2 (Alle neuen Sentinel 5-Berichte sollten diese Ansicht verwenden) ...	11-22
EVT_AGENT_RPT_V	11-26
EVT_ASSET_RPT_V	11-27
EVT_DEST_EVT_NAME_SMRY_1_RPT_V	11-28
EVT_DEST_SMRY_1_RPT_V	11-28
EVT_DEST_TXNMY_SMRY_1_RPT_V	11-29
EVT_NAME_RPT_V	11-29
EVT_PORT_SMRY_1_RPT_V	11-29
EVT_PRTCL_RPT_V	11-30
EVT_RSRC_RPT_V	11-30
EVT_SEV_SMRY_1_RPT_V	11-30
EVT_SRC_SMRY_1_RPT_V	11-31
EVT_TXNMY_RPT_V	11-31
EVT_USR_RPT_V	11-31
EXTERNAL_DATA_RPT_V	11-32
HIST_EVENTS_RPT_V	11-32
HIST_INCIDENTS_RPT_V	11-32

IMAGES_RPT_V	11-32
INCIDENTS_ASSETS_RPT_V	11-33
INCIDENTS_EVENTS_RPT_V	11-33
INCIDENTS_RPT_V	11-33
INCIDENTS_VULN_RPT_V	11-34
L_STAT_RPT_V	11-34
LOGS_RPT_V	11-35
NETWORK_IDENTITY_RPT_V	11-35
ORGANIZATION_RPT_V	11-35
PERSON_RPT_V	11-35
PHYSICAL_ASSET_RPT_V	11-36
PRODUCT_RPT_V	11-36
ROLE_RPT_V	11-36
SENSITIVITY_RPT_V	11-37
STATES_RPT_V	11-37
Ansicht UNASSIGNED_INCIDENTS_RPT_V	11-37
USERS_RPT_V	11-38
VENDOR_RPT_V	11-38
VULN_CALC_SEVERITY_RPT_V	11-39
VULN_CODE_RPT_V	11-39
VULN_INFO_RPT_V	11-39
VULN_RPT_V	11-40
VULN_RSRC_RPT_V	11-40
VULN_RSRC_SCAN_RPT_V	11-41
VULN_SCAN_RPT_V	11-41
VULN_SCAN_VULN_RPT_V	11-41
VULN_SCANNER_RPT_V	11-42
12 Sentinel-Datenbankansichten für Microsoft SQL Server	12-1
Ansichten	12-1
ADV_ALERT_CVE_RPT_V	12-1
ADV_ALERT_PRODUCT_RPT_V	12-1
ADV_ALERT_RPT_V	12-2
ADV_ATTACK_ALERT_RPT_V	12-2
ADV_ATTACK_CVE_RPT_V	12-3
ADV_ATTACK_MAP_RPT_V	12-3
ADV_ATTACK_PLUGIN_RPT_V	12-3
ADV_ATTACK_RPT_V	12-4
ADV_CREDIBILITY_RPT_V	12-4
ADV_FEED_RPT_V	12-5
ADV_PRODUCT_RPT_V	12-5
ADV_PRODUCT_SERVICE_PACK_RPT_V	12-5
ADV_PRODUCT_VERSION_RPT_V	12-6
ADV_SEVERITY_RPT_V	12-6
ADV_SUBALERT_RPT_V	12-6
ADV_URGENCY_RPT_V	12-7
ADV_VENDOR_RPT_V	12-7
ADV_VULN_PRODUCT_RPT_V	12-8
ANNOTATIONS_RPT_V	12-8
ASSET_CTGRY_RPT_V	12-8
ASSET_HOSTNAME_RPT_V	12-9
ASSET_IP_RPT_V	12-9

ASSET_LOCATION_RPT_V.....	12-9
ASSET_RPT_V.....	12-10
ASSET_VALUE_RPT_V.....	12-10
ASSET_X_ENTITY_X_ROLE_RPT_V.....	12-10
ASSOCIATIONS_RPT_V.....	12-11
ATTACHMENTS_RPT_V.....	12-11
CONFIGS_RPT_V.....	12-11
CONTACTS_RPT_V.....	12-12
CORRELATED_EVENTS_RPT_V.....	12-12
CORRELATED_EVENTS_RPT_V1.....	12-13
CRITICALITY_RPT_V.....	12-13
CUST_RPT_V.....	12-13
ENTITY_TYPE_RPT_V.....	12-14
ENV_IDENTITY_RPT_V.....	12-14
ESEC_DISPLAY_RPT_V.....	12-14
ESEC_PORT_REFERENCE_RPT_V.....	12-15
ESEC_PROTOCOL_REFERENCE_RPT_V.....	12-15
ESEC_SEQUENCE_RPT_V.....	12-16
EVENTS_ALL_RPT_V (aus Gründen der Abwärtskompatibilität angegeben).....	12-16
EVENTS_ALL_RPT_V1 (aus Gründen der Abwärtskompatibilität angegeben).....	12-21
EVENTS_RPT_V (aus Gründen der Abwärtskompatibilität angegeben).....	12-21
EVENTS_RPT_V1 (aus Gründen der Abwärtskompatibilität angegeben).....	12-22
EVENTS_RPT_V2 (aus Gründen der Abwärtskompatibilität angegeben).....	12-22
EVT_AGENT_RPT_V.....	12-26
EVT_ASSET_RPT_V.....	12-27
EVT_DEST_EVT_NAME_SMRY_1_RPT_V.....	12-28
EVT_DEST_SMRY_1_RPT_V.....	12-28
EVT_DEST_TXNMY_SMRY_1_RPT_V.....	12-29
EVT_NAME_RPT_V.....	12-29
EVT_PORT_SMRY_1_RPT_V.....	12-29
EVT_PRTCL_RPT_V.....	12-30
EVT_RSRC_RPT_V.....	12-30
EVT_SEV_SMRY_1_RPT_V.....	12-30
EVT_SRC_SMRY_1_RPT_V.....	12-31
EVT_TXNMY_RPT_V.....	12-31
EVT_USR_RPT_V.....	12-31
EXTERNAL_DATA_RPT_V.....	12-32
HIST_EVENTS_RPT_V.....	12-32
HIST_INCIDENTS_RPT_V.....	12-32
IMAGES_RPT_V.....	12-32
INCIDENTS_ASSETS_RPT_V.....	12-33
INCIDENTS_EVENTS_RPT_V.....	12-33
INCIDENTS_RPT_V.....	12-33
INCIDENTS_VULN_RPT_V.....	12-34
L_STAT_RPT_V.....	12-34
LOGS_RPT_V.....	12-35
NETWORK_IDENTITY_RPT_V.....	12-35
ORGANIZATION_RPT_V.....	12-35
PERSON_RPT_V.....	12-35
PHYSICAL_ASSET_RPT_V.....	12-36
PRODUCT_RPT_V.....	12-36
ROLE_RPT_V.....	12-36

SENSITIVITY_RPT_V	12-37
STATES_RPT_V	12-37
Ansicht UNASSIGNED_INCIDENTS_RPT_V	12-37
USERS_RPT_V	12-38
VENDOR_RPT_V	12-38
VULN_CALC_SEVERITY_RPT_V	12-38
VULN_CODE_RPT_V	12-39
VULN_INFO_RPT_V	12-39
VULN_RPT_V	12-39
VULN_RSRC_RPT_V	12-40
VULN_RSRC_SCAN_RPT_V	12-41
VULN_SCAN_RPT_V	12-41
VULN_SCAN_VULN_RPT_V	12-41
VULN_SCANNER_RPT_V	12-42

A Checkliste für die Sentinel-Fehlersuche..... A-1

B Einrichten des Sentinel-Service-Anmeldekontos als

NT AUTHORITY\NetworkService B-1

So richten Sie NT AUTHORITY\NetworkService als Anmeldekonto für den Sentinel-Service ein	B-3
Hinzufügen eines Sentinel-Service als Anmeldekonto zu ESEC- und ESEC_WF-DB-Instanzen	B-3
Ändern des Anmeldekontos des Sentinel-Service in NT AUTHORITY\NetworkService	B-8
Gewährleisten des erfolgreichen Starts des Sentinel-Services	B-9

C Benutzer, Funktionen und Zugriffsberechtigungen der Sentinel-Datenbank..... C-1

Sentinel-Datenbankinstanz	C-1
ESEC	C-1
ESEC_WF	C-1
Benutzer der Sentinel-Datenbank	C-2
Zusammenfassung	C-2
esecadm	C-2
esecapp	C-2
esecdba	C-2
esecrpt	C-2
Funktionen in der Sentinel-Datenbank	C-2
Zusammenfassung	C-2
ESEC_APP	C-3
ESEC_ETL	C-9
ESEC_USER	C-12
Sentinel Server-Funktionen	C-14
Windows-Domänenauthentifizierung: DB-Benutzer und -Berechtigungen	C-14

D Tabellen mit Sentinel-Serviceberechtigungen D-1

Sentinel Server (Correlation Engine) D-1

Collector Manager D-2

Sentinel Communication D-5

Datenbankserver (ohne DAS) D-6

Datenbankserver (mit DAS) D-7

Reporting Server D-9

1

Sentinel™ 5: Einführung zum Referenzhandbuch für Benutzer

HINWEIS: Die Begriffe „Agent“ und „Collector“ sind austauschbar. Im Folgenden werden Agenten als „Collectors“ bezeichnet.

Das Sentinel-Referenzhandbuch für Benutzer dient als Referenz für folgende Aspekte:

- Wizard-Skriptsprache
- Wizard-Parsing-Befehle
- Wizard-Administratorfunktionen
- META-Tags für Wizard und Sentinel
- Benutzerberechtigungen für die Sentinel Console
- Sentinel Correlation Engine
- Sentinel-Befehlszeilenoptionen
- Sentinel Server-Datenbankansichten

In diesem Handbuch wird davon ausgegangen, dass Sie mit den Aspekten der Netzwerksicherheit, der Datenbankverwaltung sowie den UNIX-Betriebssystemen vertraut sind.

Inhalt

Dieses Handbuch enthält folgende Kapitel:

- Kapitel 1 – Sentinel: Einführung zum Referenzhandbuch
- Kapitel 2 – Wizard-Skriptsprache
- Kapitel 3 – Wizard-Parsing-Befehle
- Kapitel 4 – Wizard-Administratorfunktionen
- Kapitel 5 – META-Tags für Wizard und Sentinel
- Kapitel 6 – Benutzerberechtigungen für Sentinel Control Center
- Kapitel 7 – Sentinel Correlation Engine
- Kapitel 8 – Sentinel-Korrelations-Befehlszeilenoptionen
- Kapitel 9 – Sentinel Data Access Service
- Kapitel 10 – Standardbenutzerpasswörter ändern
- Kapitel 11 – Sentinel-Datenbankansichten für Oracle
- Kapitel 12 – Sentinel-Datenbankansichten für Microsoft SQL Server
- Anhang A – Checkliste für die Sentinel-Fehlersuche
- Anhang B – Einrichten des eSecurity-Service-Anmeldekontos als NT AUTHORITY\NetworkService
- Anhang C – Benutzer, Funktionen und Zugriffsberechtigungen der Sentinel-Datenbank
- Anhang D – Tabellen mit Sentinel-Serviceberechtigungen

Verwendete Konventionen

Hinweise und Warnhinweise

HINWEIS: Hinweise stellen zusätzliche Informationen bereit, die sich als hilfreich erweisen können.

ACHTUNG: Warnhinweise stellen zusätzliche Informationen bereit, mit denen sich Beschädigungen des Systems bzw. Datenverluste u. U. vermeiden lassen.

Befehle

Befehle sind in Courier-Schriftart angegeben. Beispiel:

```
useradd -g dba -d /export/home/oracle -m -s /bin/csh
oracle
```

Weitere Sentinel-Referenzen

Folgende Handbücher sind auf den Sentinel-Installations-CDs enthalten:

- Sentinel™ 5-Installationshandbuch
- Sentinel™ 5-Benutzerhandbuch
- Sentinel™ 5 Wizard-Benutzerhandbuch
- Sentinel™ 5-Referenzhandbuch für Benutzer
- Sentinel™5-Handbuch für Drittanbieter-Integration
- Versionshinweise

Kontaktaufnahme mit Novell

- Website: <http://www.novell.com>
- Technischer Support von Novell: <http://www.novell.com/support/index.html>
- Internationaler technischer Support von Novell:
http://support.novell.com/phone.html?sourceidint=suplnav4_phonesup
- Self-Support:
http://support.novell.com/support_options.html?sourceidint=suplnav_supportprog
- Für Support rund um die Uhr: +1 800-858-4000

2

Wizard-Skriptsprache

HINWEIS: Die Begriffe „Agent“ und „Collector“ sind austauschbar. Im Folgenden werden Agenten als „Collectors“ bezeichnet.

In diesem und im nachfolgenden Kapitel wird die Verwendung der Wizard-Skriptsprache zur Erstellung von Skripts erläutert. Operatoren in den unterschiedlichen Zeichenketten und Parsing-Befehle (Analysebefehle), die bei der Collector-Erstellung verwendet werden, werden hier abgedeckt.

Folgende Punkte werden erläutert:

- [decide-Zeichenketten](#)
- [Reguläre Ausdrücke](#)

decide-Zeichenketten

Bei Zeichenketten muss die Groß- und Kleinschreibung beachtet werden.

Im Rahmen des Polling-Vorgangs von Collectors werden im internen Empfangspuffer verschiedene Informationen gesammelt. Zeichenketten vom decide-Typ geben an, dass eine Entscheidung hinsichtlich der im internen Puffer eingegangenen und gespeicherten Daten getroffen wird. Eine decide-Zeichenkette wird entweder als „true“ (wahr) oder „false“ (falsch) ausgewertet. Wenn ein Syntaxfehler vorliegt bzw. wenn das Feld für den decide-Typ frei gelassen wird, ist die Entscheidung „false“.

Die decide-Zeichenkette wird nur ausgewertet, wenn als decide-Typ eine Zeichenkette bzw. Daten angegeben sind.

Bearbeiten des Rx Buffer Pointer (Zeiger des Empfangspuffers)

Jeder Port in Wizard weist einen eigenen Receive Buffer Pointer (Empfangspufferzeiger) auf. Der Receive Buffer Pointer zeigt auf Datenbyte im Empfangspuffer. Vor den einzelnen ausgewerteten decide-Zeichenketten wird der Receive Buffer Pointer auf seinen verwalteten Wert zurückgesetzt (normalerweise 0, es sei denn, er wurde durch eine Entscheidung verändert, bei der der (:)-Suchoperator zum Einsatz kam).

- 0 zeigt auf kein Byte im Empfangspuffer
- 1 zeigt auf das erste Datenbyte, 2 zeigt auf das zweite Datenbyte usw.

Format

Eine decide-Zeichenkette besteht aus einer Sequenz logischer Operatoren (LO) und regulären Ausdrücken.

Logische Operatoren und Zeichenkettenoperatoren müssen nicht in jeder Sequenz vorhanden sein. Hier einige Regeln zu ihrer Verwendung:

- Logische Operatoren erstellen in der decide-Zeichenkette boolesche Ausdrücke („true“ oder „false“) und werden basierend auf folgendem Vorrang ausgewertet:
~ Not
& And
- Ein Zeichenkettenoperator gibt eine Reihe von Zeichen an, nach denen im Empfangspuffer gesucht werden soll. Mithilfe des Zeichenkettenoperators wird ab der Receive Buffer Pointer Position vorwärts ein Byte-für-Byte-Suchvorgang durchgeführt.

HINWEIS: Da das Feld für den decide-Typ beim letzten druckbaren Zeichen abgeschnitten wird, muss das Hex-Äquivalent eines Leerschritts verwendet werden. Der :-Operator darf nicht mit dem NULL-Operator verwendet werden.

Parameternamen

Zur Angabe eines Parameters in einer decide-Zeichenkette muss der Parametername in geschweifte Klammern ({ }) gesetzt werden. Bei der Erstellung des Skripts werden der Parametername und die geschweiften Klammern durch den Wert des Parameters ersetzt.

Wenn der angegebene Parametername in der Parameterdatei, anhand der das Skript erstellt wird, nicht vorhanden ist, verbleiben der Parameternamenausdruck und die geschweiften Klammern in den decide-Zeichenkettendaten.

Parameternamenausdrücke können sich an beliebiger Stelle in der decide-Zeichenkette befinden. Sie dürfen jedoch nicht verschachtelt werden (in sich selbst einen weiteren Parameternamenausdruck enthalten).

Vorgangshierarchie in einer decide-Zeichenkette

Jeder Vorgang in einer decide-Zeichenkette wird entweder als „true“ (1) oder „false“ (0) ausgewertet. Die Bearbeitung von Vorgängen in einer decide-Zeichenkette erfolgt stets in der durch die logische Operatorsyntax bestimmten Reihenfolge.

- Wenn mehrere Vorgänge verwendet werden, erfolgt die Zeichenkettenauswertung von links nach rechts.
- Wenn Klammern verwendet werden, wird der logische Operator innerhalb der einzelnen Klammernpaare zuerst ausgewertet.
- Die nächsten auszuwertenden Vorgänge sind „not“ (~), „and“ (&).

Auch bei der Verwendung der Zeichenkettenoperatorsyntax gilt eine bestimmte Vorgangsreihenfolge:

- Der zurückgesetzte Rx Buffer Pointer (Empfangspufferzeiger) wird zuerst ausgewertet.
- Alle anderen Syntaxzeichen haben denselben Vorrang und werden gemäß ihrer Reihenfolge, von links nach rechts, ausgewertet.

Regeln für den Receive Buffer Pointer

Der Wert des Receive Buffer Pointer wird durch folgende Regeln bestimmt:

- Wenn die Suche nach einer Reihe von Zeichen erfolgreich ist, gilt der Suchvorgang als „true“ und der Receive Buffer Pointer wird auf das erste Byte in der Zeichenkette gesetzt, das gefunden wurde.

decide-Zeichenkette: DE

A BCDE F GH

^

A BCDE F GH

^

- Wenn die Suche nach einer Reihe von Zeichen nicht erfolgreich ist, gilt der Suchvorgang als „false“ und der Receive Buffer Pointer wird auf den verwalteten Wert zurückgesetzt.

decide-Zeichenkette: DEJ

A BCDE F GH

^

A BCDE F GH

^

Prüfen auf leeren Empfangspuffer

Verwenden Sie zum Prüfen auf einen leeren Empfangspuffer folgende decide-Zeichenkette:

NULL

Beispiel für Auswertungen und Ergebnisse von decide-Zeichenketten

Alphanumerische decide-Zeichenketten

Nachfolgend sind alphanumerische decide-Zeichenketten für einen Beispielpuffer aufgeführt:

ABCDEFGH IJKLMNOP (line feed) YZ<[&

decide-Zeichenkette	Logischer Ausdruck	Ergebnis
A	1	1
P	0	0
\41\ (HEX für A)	1	1
AB	1	1
\4142\ (HEX für AB)	1	1
ABD	0	0
A&B	1 & 1	1
A&P	1 & 0	0
A+P	1 + 0	1
A\42\ (HEX für B)	1	1
A&BC	1 & 1	1
DEF&ABC	1 & 0	0
ABC&DEF	1 & 1	1
ABC&BCD	1 & 1	1
ABC&ABC	1 & 0	0

decide-Zeichenkette	Logischer Ausdruck	Ergebnis
\OA\ (HEX für line feed (Zeilenvorschub))	1	1
NULL *	0	0

Wenn im Empfangspuffer keine Zeichen gefunden werden, ist das Ergebnis TRUE.

HEX-decide-Zeichenketten

Nachfolgend sind HEX-decide-Zeichenketten für einen Beispielpuffer (HEX) aufgeführt:

02 0A 10 FF 1F 2E 3C 03

decide-Zeichenkette	Logischer Ausdruck	Ergebnis
\020A\&\FF\	1 & 1	1
\02\	0	0
\02\&\03\	1 & 1	1
\03\&\02\	1 & 0	0

Reguläre Ausdrücke

Sonderzeichen und Sequenzen von Zeichen, die in Schreibmustern für regelmäßige Ausdrücke verwendet werden.

Sentinel verwendet eine mit POSIX (Portable Operating System Interface for UNIX) kompatible Bibliothek für reguläre Ausdrücke. POSIX setzt sich aus IEEE-(Institute of Electrical & Electronics Engineers-) und ISO-(Industry Standards Organisation-)Standards zusammen, mit denen die Kompatibilität zwischen POSIX-fähigen Betriebssystemen gewährleistet wird; hierzu zählt der Großteil der UNIX-Varianten.

Zusammenfassung von Sonderzeichen für reguläre Ausdrücke

In der nachfolgenden Tabelle sind die Sonderzeichen zusammengefasst, die in regulären Ausdrücken für die Funktionen SEARCH (Suchen) und REPLACE (Ersetzen) verwendet werden können.

Zeichen	Verwendung/Beispiel
\	Markiert das nächste Zeichen als Sonderzeichen. n gleicht das Zeichen „n“ ab. Die Sequenz \n stimmt mit einem Zeilenvorschub oder newline-Zeichen (Ende der Zeile) überein, damit jedoch das "\" den Parser durchläuft, muss ihm das Ausnahmezeichen "/" vorangestellt werden; für die Übergabe von \n muss folglich \n verwendet werden.
^	Gleicht den Beginn der Eingabe oder Zeile ab.
\$	Gleicht das Ende der Eingabe oder Zeile ab.
*	Gleicht das vorangegangene Zeichen nullmal oder mehrmals ab. go* gleicht entweder „g“ oder „goo“ ab.
+	Gleicht das vorangegangene Zeichen einmal oder mehrmals ab. go+ gleicht „goo“, aber nicht „g“ ab.
?	Gleicht das vorangegangene Zeichen nullmal oder einmal ab. a?te? gleicht das „te“ in „eater“ ab.
.	Gleicht ein beliebiges einzelnes Zeichen mit Ausnahme eines newline-Zeichens (Ende der Zeile) ab.
x y	Gleicht entweder x oder y ab. z good? gleicht „goo“ oder „good“ oder „z“ ab.

Zeichen	Verwendung/Beispiel
{n}	n ist eine nicht negative Ganzzahl. Gleicht genau n-Mal ab. e{3} gleicht das „e“ in „Ted“ nicht ab, jedoch die ersten drei „e“ in „greeeeeed“ ab.
{n,}	n ist eine nicht negative Ganzzahl. Gleicht mindestens n ab. e{3,} gleicht das „e“ in „Ted“ nicht ab und gleicht alle „e“ in „greeeeeed“. e{1,} ist äquivalent zu e+.
{n,m}	m und n sind nicht negative Ganzzahlen. Gleicht mindestens n- und höchstens m-Mal ab. e{1,3} gleicht die ersten drei „e“ in „greeeeeed“ ab.
[xyz]	Ein Zeichensatz. Gleicht jedes der eingeschlossenen Zeichen ab. [xyz] gleicht das „y“ in „play“ ab.
[^xyz]	Ein negativer Zeichensatz. Gleicht sämtliche nicht eingeschlossenen Zeichen ab. [^xyz]/ gleicht das „v“ in „vain“ ab.
[0-9]	Gleicht eine Ziffer ab.
[^0-9]	Gleicht ein anderes Zeichen als eine Ziffer ab.
[A-Za-z0-9_]	Gleicht ein beliebiges Wort ab, einschließlich eines Unterstrichs.
[^A-Za-z0-9_]	Gleicht ein anderes Zeichen als ein Wort ab.
/n/	Gleicht n ab, wobei n ein Oktal-, Hexadezimal- oder Dezimal-Escape-Wert ist. Ermöglicht die Einbettung von ASCII-(American Standard Code for Information Interchange)-Codes in reguläre Ausdrücke.

Leerzeichen in regulären Ausdrücken

In regulären Ausdrücken bestehen Leerzeichen aus einem oder mehreren Leerschritten, bei denen es sich um eines der folgenden Zeichen handeln kann:

Symbolischer Name	UCS	Beschreibung
<tab> (Tabulator)	<U0009>	ZEICHENTABULIERUNG (HT)
<carriage-return> (Zeilenschaltung)	<U000D>	ZEILENSCHALTUNG (CR)
<newline> (Neue Zeile)	<U000A>	ZEILENVORSCHUB (LF)
<vertical-tab> (Vertikaltabulator)	<U000B>	ZEILENTABULIERUNG (VT)
<form-feed> (Formularvorschub)	<U000C>	FORMULARVORSCHUB (FF)
<space> (Leerschritt)	<U0020>	LEERSCHRITT

Parsing-Befehle

Die Wizard-Parsing-Sprache ist funktionsorientiert. Mit dem Großteil der Parsing-Funktionen können Sie Wizard-Variablen und deren Inhalt bearbeiten. Die Wizard-Parsing-Sprache unterstützt vier Variablentypen:

- Ganzzahlvariable (der Variablenname beginnt mit i)
- Float-Variable (der Variablenname beginnt mit f)
- Zeichenkette mit variabler Länge (der Variablenname beginnt mit einem beliebigen Zeichen außer i und f)
- Arrays mit Variablen (der Variablenname endet mit []). Bei Arrayvariablentypen kann es sich um Arrays von Ganzzahlen, Float-Variablen oder Zeichenketten handeln

Diese Variablen sind für die einzelnen Wizard-Ports lokal und werden nicht global für alle Wizard-Ports freigegeben. Mithilfe von Parsing-Befehlen können Daten aus dem Empfangspuffer in Zeichenkettenvariablen kopiert werden.

Der Empfangspuffer enthält die Daten, die von Wizard-Kommunikationsport, -Socket-Port, -Datei oder -Vorgang übermittelt wurden.

Die Länge der zu kopierenden Byte sowie die Position, ab der die Byte kopiert werden sollen, können mithilfe folgender Parsing-Befehle gesteuert werden:

- SEARCH()
- SKIP()
- SKIPWORD()
- NEGSEARCH()
- RESET()
- COPY()

Daten aus dem Empfangspuffer können mit dem APPEND-Befehl an eine Zeichenkette angefügt werden. Mithilfe der Wizard-Parsing-Sprache können zudem Daten aus Zeichenkettenvariablen in andere Zeichenkettenvariablen kopiert bzw. an diese angefügt werden.

Einfache Datentypen

number

Ziffern darf nur beim SKIP-Befehl, beim SKIPWORD-Befehl und beim SET-Befehl ein + oder - vorangestellt werden. Beispiel:

```
0, 10, 2.5
```

ivar (Ganzzahlvariablen)

Als Ganzzahlvariablen werden 32-Bit-Zahlen mit Vorzeichen bezeichnet. Der Variablenname muss mit einem I oder i beginnen. Beispiel:

```
i_count, I_severity, i, i[55], i[index]
```

Bei der Ganzzahlvariablen, i[55], handelt es sich um den 55. Index für das Ganzzahlarray, i[]. Beim Index für ein Array kann es sich auch um eine Ganzzahlvariable handeln.

fvar (Float-Variablen)

Als Float-Variable werden 32-Bit-Fließkommazahlen bezeichnet. Der Variablenname muss mit einem F oder f beginnen. Beispiel:

```
f_rate, F_queue, f, f[1], f[index]
```

svar (Zeichenkettenvariablen)

Zeichenkettenvariablen enthalten Zeichenketten variabler Länge.

Zeichenkettenvariablenamen dürfen nicht mit einem I, i, F oder f beginnen. Beispiel:

```
resource, date, _message, string[1000], string[i_sev]
```

array (Variablenarrays)

Variablenarrays können für Arrays von Variablen vom Typ „ivar“, „fvar“ und „svar“ stehen. Beispiel:

```
i_bits[], F_values[], s_resources[]
```

Arrays können mit einem beliebigen numerischen Index indiziert werden, ohne dass Speicherplatz verschwendet wird. Der Zugriff auf ivar[1000] ist nicht gleichbedeutend mit der Zuordnung von Arbeitsspeicher für 1.000 Ganzzahlvariablen.

Eine indizierte Arrayvariable wird wie jede andere Variable (ivar, svar und fvar) gehandhabt.

Hier ein Beispiel für zulässige Syntax für den POPUP-Befehl:

```
POPUP(xterm_display[4], data[i_count])
```

Quoted Data

Quoted Data (Daten in Anführungszeichen) werden wie folgt abgesucht und analysiert:

- /=Ausnahmezeichen: Byte, das auf / folgt, ungeachtet eventueller Sonderbedeutung aufnehmen; wenn eines der Sonderzeichen in der Zeichenkette verwendet werden soll, muss / diesem Zeichen vorangestellt werden. So wird beispielsweise corp\router für corp\router verwendet
- \xx x xx=Hex-Daten (ein oder zwei Zeichen pro Byte möglich): \0ad\, \0a0d\, \a d\, \0a 0d\ und \0a d\ stehen alle für Zeilenvorschub/Zeilenschaltung

Alle anderen Zeichen werden direkt angegeben.

Derived Aggregate-Datentypen

In der nachfolgenden Tabelle werden abgeleitete Aggregat-Datentypen aufgelistet:

Typ	Beschreibung
Beliebig	Zahl, ivar, fvar, svar, Anführungszeichen
Numerisch	Zahl, ivar, fvar, ivar[index], fvar[index]
Zeichenkette	svar, svar[index], Anführungszeichen
Variable	ivar, fvar, svar, ivar[index], fvar[index], svar[index]
numvar	ivar, fvar, ivar[index], fvar[index]
Array	ivar[], fvar[], svar[]
numvar-Array	ivar[], fvar[]
Zeichenkettenvariablenarray	svar[]

Sonderregeln für Variablen

Nachfolgend sind Sonderregeln für Variablen aufgeführt.

- Bei Variablennamen muss die Groß- und Kleinschreibung beachtet werden
- Wenn eine numvar erstmals (mit Ausnahme der Fälle, in denen ihr Wert festgelegt wird) verwendet wird, wird sie auf 0 eingestellt
- Wenn eine svar erstmals (mit Ausnahme der Fälle, in denen ihr Wert festgelegt wird) verwendet wird, wird sie auf null ("") eingestellt
- Eine indiziertes Array wird wie jede andere Variable des jeweiligen Typs (ivar, fvar oder svar) gehandhabt

- Wenn ein oder mehr Parsing-Befehle auskommentiert werden sollen oder wenn Kommentare in den Parsing-Text aufgenommen werden sollen, müssen die Kommentare in `/* */` gesetzt werden

Beispiel:

```
/* this is a comment */  
/* these are commented out parsing commands  
COPY(s: "test")  
DISPLAY()  
*/
```

3

Wizard-Parsing-Befehle

HINWEIS: Die Begriffe „Agent“ und „Collector“ sind austauschbar. Im Folgenden werden Agenten als „Collectors“ bezeichnet.

In diesem Kapitel werden die Wizard-Parsing-Befehle, also die Befehle für Analysevorgänge, die bei der Collector-Erstellung zum Einsatz kommen, alphabetisch aufgelistet. Nachfolgend finden Sie eine Liste der nach Funktion geordneten Parsing-Befehle.

Funktion	Parsing-Befehl
Datenbankinteraktion	DBCLOSE DBDELETE DBGETROW DBINSERT DBOPEN DBSELECT
Durchführen der Fehlersuche	BREAKPOINT DISPLAY POPUP
Dateiinteraktion	FILEA FILEL FILER FILEW
Logische Vorgänge	COMPARE ELSE ENDFOR ENDIF ENDWHILE FOR IF LOOKUP WHILE
Netzwerkinteraktion	SOCKETW
Benachrichtigung	ALERT CLEARTAGS CONSTANTTAGS EVENT INDICATOR PAUSE

Funktion	Parsing-Befehl
Bearbeitung von Rohdaten	BITFIELD BYTEFIELD CONVERT CRC DECODE DECODEMIME ENCODE ENCODEMIME HEXTONUM NUMTOHEX SETBYTES STRIP STRIP-ASCII-RANGE
Bearbeitung von Zeichenketten	APPEND COPY COPY-FROM-RX-BUFF-UNTIL-SEARCH COPY-FROM-RX-BUFF COPY-FROM-STRING-TO-STRING-UNTIL-SEARCH COPY-STRING-TO-STRING LENGTH LENGTH-OPTION2 NEGSEARCH PARSER_ATTACHVARIABLE PARSER_CREATEBASIC PARSER_NEXT PARSER_PARSESTRING PRINTF REGEXPREPLACE REGEXPSEARCH REGEXPSEARCH_EXPLICIT REGEXPSEARCH_STRING REPLACE SEARCH SKIP SKIPWORD STONUM TOKENIZE TOLOWER TOUPPER TOKENIZE TRANSLATE
Dienstprogramm	DATE DATETIME PAUSE SHELL TBOSSETCOMMAND TBOSSETREQUEST TIME

Funktion	Parsing-Befehl
Verarbeitung von Variablen	CLEAR DELETE GETCONFIG GETENV INC RESET RXBUFF SET SETCONFIG
Absuchen auf Anfälligkeit	INFO_CLEARTAGS INFO_CLOSE INFO_CONSTANTTAGS INFO_CREATE INFO_DUMP INFO_PUSH INFO_SEND INFO_SETTAG

Befehlsformat und Verwendung von Arrays

In Parsing-Befehlsformaten wird anhand bestimmter Symbole eine bestimmte Bedeutung vermittelt. Nachfolgend finden Sie Beispiele für diese Symbole:

Beispiel des verwendeten Symbols	Beispiel für die Bedeutung des Symbols
[parameter]	Eckige Klammern kennzeichnen optionale Parameter.
<parameter>	Spitze Klammern kennzeichnen erforderliche Parameter, die von Ihnen angegeben werden.
a	Hier muss „a“ genau so eingegeben werden
a b	Verwenden Sie entweder „a“ oder „b“ genau wie angegeben, aber nicht beides
<element> ::= <definition>	„element“ kann durch „definition“ ersetzt werden
<varList> Hierbei gilt: <varList> ::= var [, <varList>]	Wird für rekursive Definitionen zur Beschreibung einer Liste mit Variablen verwendet, in der mindestens eine Variable erforderlich ist
...	Die Wiederholung vorangegangener Parameter ist zulässig.
/	Der Schrägstrich dient als „Escape“-Zeichen und ermöglicht die Verwendung von Sonderzeichen, beispielsweise dem umgekehrten Schrägstrich (\).

Arrays sind in Ausdrücken zulässig. Beispiel:

Angabe	Entsprechungen
SET(i_var = 2)	i_arr[3]
SET(i_arr[3]=2)	i_arr[i_var] i_arr[1+2] i_arr[1+1_var] i_arr[i_arr[3]]

Befehle

ALERT



Mit dem ALERT-Befehl werden Ereignismeldungen an Sentinel weitergeleitet.

- Der erste erforderliche Parameter definiert den Ressourcennamen
- Der zweite erforderliche Parameter definiert den Text der Ereignismeldung
- Der dritte erforderliche Parameter definiert den Schweregrad des Ereignisses
- Datum und Uhrzeit der Ereignismeldung können als optionale Parameter definiert werden
 - Der date-Parameter kann alleine verwendet werden
 - Der time-Parameter muss gemeinsam mit dem date-Parameter verwendet werden

Format

```
ALERT(resource, message, iseverity)
```

oder

```
ALERT(resource, message, iseverity[, date[, time]])
```

Der date-Parameter kann nur in Kombination mit dem time-Parameter verwendet werden.

HINWEIS: Verwenden Sie den STONUM-Befehl, um „iseverity“ von einer Zeichenkette in eine Ganzzahl umzuwandeln.

Datentypen

Argument	Typ	Beschreibung
Ressource	Zeichenkette (EINGABE)	Die Ressource und optional die Teilressource, an die ein Ereignis gesendet wird (Beispiel: xterm:tcp_retransmits).
Meldungszeichenkette	(EINGABE)	Der Meldungstext für die Ereignismeldung.
iseverity	Numerisch (EINGABE)	Die numerische Darstellung der Priorität dieser Ereignismeldung (0–5). 0 = Informativ 1 = Hinweis 2 = Warnmeldung 3 = Geringfügig 4 = Erheblich 5 = Kritisch
Datumszeichenkette	(EINGABE) [OPTIONAL]	Legt das Datum der Ereignismeldung im Format MM-TT-JJJJ fest (Beispiel: „12-01-2002“) (Standard = heutiges Datum).

Argument	Typ	Beschreibung
Uhrzeitzeichenkette	(EINGABE) [OPTIONAL]	Legt die Uhrzeit der Ereignismeldung im Format HH:MM:SS fest (Beispiel: „15:14:34“) (Standard = aktuelle Uhrzeit); muss gemeinsam mit dem date-Parameter verwendet werden.

Beispiel:

```
ALERT("xterm:tcp_retransmits", msg_txt, ivar[3])
ALERT("router_subnet_15", msg_txt, "c")
ALERT(resource, "Server not responding", iseverity)
ALERT("Mux184:card1", "C1 not funct. properly.", 4)
ALERT("Firewall", "Connection lost to Firewall.", 5)
ALERT("CB5", "Channel Bank 5 being serviced", "Maint")
ALERT(resource, message, isev, thedate, thetime)
ALERT("Switch3", oos_msg, 5, "07-30-1997", "07:03:23")
```

APPEND



Der APPEND-Befehl fügt einer Zeichenkettenvariablen Daten aus dem Empfangspuffer, aus einer Zeichenkettenvariable oder aus einer Zeichenkette mit Anführungszeichen hinzu. Es gilt Folgendes:

- Sämtliche APPEND-Parameter sind optional, mit Ausnahme des Parameters für das Ziel
- Das Ziel für die Daten (Zeichenkettenvariable) kann mithilfe der APPEND-Parameter angegeben werden
- Ein Offset in die Quelle kann angegeben werden, um zu steuern, wohin Daten aus den Quelldaten kopiert werden
- Die Anzahl der Byte, die an die Zielvariable angefügt werden sollen, kann mithilfe des Längenparameters (ilen) angegeben werden; anderenfalls wird standardmäßig die Länge der Quelldaten verwendet
- Zusätzlich zur Angabe eines numerischen Längenparameters kann die Länge mithilfe einer Zeichenkette definiert werden
- Wenn eine Zeichenkette als Längenparameter verwendet wird, muss es sich beim Quellparameter entweder um den Empfangspuffer oder eine svar handeln
- Durch die Verwendung einer Zeichenkette als Längenparameter fügt die Collector-Engine Byte aus den Quelldaten (beginnend bei Offset) an die Zielvariable an, und zwar bis zum, aber ohne das erste Zeichen der Zeichenkette (wenn sie gefunden wird) (wenn die Zeichenkette nicht gefunden wird, werden keine Byte angefügt)
- Wenn der Offset- bzw. Längenparameter außerhalb des Bereichs der Quellvariable liegen, werden so viele Byte wie möglich angefügt, bis zum Ende der Quelldaten
- Wenn der Offset der Länge der Quelldaten entspricht bzw. diese übersteigt, werden keine Byte an die Zielvariable angefügt (wenn kein Offset angegeben wird, wird standardmäßig Null verwendet)

Format

```
APPEND(<dest>: [source] [, [search] [, [ilen] [,
        [ioffset] ]])
APPEND(<dest>: [source] [, [ilen] [, [ioffset] ]])
APPEND(<dest>: [ilen] [, [offset]])
```

Datentyp

Argument	Typ	Beschreibung
dest	svar (AUSGABE)	Die Datenzeichenkettenvariable, an die Byte angefügt werden.
source	Zeichenkette (EINGABE) [OPTIONAL] oder svar	Die Zeichenkette mit den Quellbyte, die an die Zielzeichenkette angefügt werden sollen. (Standard = Empfangspuffer) Wenn der Suchparameter verwendet wird.
search	Zeichenkette (EINGABE) [OPTIONAL]	Eine Zeichenkette, mit der Folgendes angegeben wird: Kopieren bis zu den Byte, nach denen in der Quellzeichenkette gesucht werden soll.
ilen	Numerisch (EINGABE) [OPTIONAL]	Die Anzahl der Byte, die aus der Quelle an das Ziel angefügt werden sollen.
ioffset	Numerisch (EINGABE) [OPTIONAL]	Der Offset in der Quelle, bei dem mit dem Anfügen von Daten begonnen werden soll.

In den nachfolgenden Beispielen werden Byte aus dem Empfangspuffer an eine Ziel-svar (dest) angefügt. Die Rx Buffer Pointer Position, also die Position des Empfangspufferzeigers, wird dem Offset-Wert hinzugefügt, um die erste Position der Daten für das Anfügen anzugeben. Das ^-Symbol kennzeichnet die Rx Buffer Pointer Position.

```
APPEND(svar:ilen)
APPEND(svar:3)
APPEND(svar:,ioffset)
APPEND(source:ilen,ioffset)
APPEND(svar: 10, 12)
```

Das obige Beispiel basiert auf folgenden Annahmen.

```
rxbuff="receive buffer"
^ (Rx buffer pointer position)
dest="A destination string"
source="A source string"
ilen=3
ioffset=3
```

Geben Sie Folgendes ein:

```
APPEND(dest:)
```

Ergebnis:

```
dest = "A destination stringreceive buffer"
```

Wenn jedoch Folgendes eingegeben wurde:

```
APPEND(dest:ilen)
```

Ergebnis:

```
dest = "A destination stringrec"
```

Wenn jedoch Folgendes eingegeben wurde:

```
APPEND(dest:,ioffset)
```

Ergebnis:

```
dest = "A destination stringreceive buffer"
```

In den nachfolgenden Beispielen werden Byte aus dem Empfangspuffer an eine Ziel-svar (dest) angefügt, und zwar bis zur, jedoch ohne die Suchzeichenkette. Wenn im Empfangspuffer keine Suchzeichenkette gefunden wird (nach Rx Buffer Pointer + Offset Position), werden keine Byte angefügt.

Geben Sie Folgendes ein:

```
APPEND(dest:,"buffer")
```

Ergebnis:

```
dest = "A destination stringreceive "
```

Geben Sie Folgendes ein:

```
APPEND(dest:,"buffer", 9)
```

Ergebnis:

```
dest = "A destination string"
```

Die nachfolgenden Beispiele dienen dem Anfügen einer Teilzeichenkette aus dem Empfangspuffer, wobei Folgendes angenommen wird:

```
Rx Buffer = "Minor Alarm Firewall A"
```

Geben Sie Folgendes ein:

```
COPY(message:"Resource Name is: ")
```

```
APPEND(message:,6)
```

Ergebnis:

```
message = "Resource Name is: Alarm Firewall A"
```

BITFIELD



Mit dem BITFIELD-Befehl werden Byte in Bit umgewandelt. Dieser Befehl wandelt jedes Byte in einer Zeichenkette beliebiger Länge durch Einfügen in ein Ganzzahl-Array, ein Float-Array oder eine Zeichenkette in 8 Bit um (0 oder 1) um.

ACHTUNG: Die Ausgabe ist 8 Mal größer als die Ausgabe, folglich kann sich der Bitfield-Parsing-Befehl bei unsachgemäßer Verwendung als sehr speicherintensiv erweisen. Dies ist beispielsweise der Fall, wenn Eingabezeichenketten eine sehr große Anzahl an Byte enthalten.

Format

BITFIELD(s_bytes, dest_var)

Datentypen

Argument	Typ	Beschreibung
s_bytes	Zeichenkette (EINGABE)	Beliebige Anzahl von ASCII- oder Hex-Byte in einer Zeichenkette.
dest_var	numvar-Array (AUSGABE)	Array mit Ganzzahlen (auf 0 oder 1 eingestellt). Die Anzahl der Bit entspricht der Anzahl der Byte in s_bytes multipliziert mit 8. Für jeden 8-Bit-Satz erfolgt die Sortierung von Most Significant Bit (MSB) nach Least Significant Bit (LSB). Beispiel: idest_var[0] = MSB of Byte 1 idest_var[1] = Next MSB of Byte 1 idest_var[2] = Next MSB of Byte 1 idest_var[3] = Next MSB of Byte 1 idest_var[4] = Next MSB of Byte 1 idest_var[5] = Next MSB of Byte 1 idest_var[6] = Next MSB of Byte 1 idest_var[7] = LSB of Byte 1 idest_var[8] = MSB of Byte 2 idest_var[9] = Next MSB of Byte 2 ... idest_var[n * 8 - 1] = LSB of Byte n
	Oder svar (AUSGABE)	Eine Zeichenkette, die ein Mehrfaches von 8 Byte enthält; hierbei steht jedes Byte für ein Bit in den eingegebenen Byte. Die Byte in dieser Zeichenkette sind stets auf ASCII 0 oder 1 eingestellt. Für sämtliche aufeinanderfolgenden 8 Bit in den einzelnen Zeichenketten erfolgt die ASCII-Sortierung (0er und 1er) von MSB nach LSB. Beispiel: Wenn s_bytes = "\5AFE\ Dann dest_var= "010110101111110"

HINWEIS: Beim zweiten Parameter für Bitfield (`dest_var`) muss es sich um eine Zeichenkette handeln (z. B. `ivar[]` oder `fvar[]`).

Beispiel:

```
BITFIELD("\00\", f_bit_array[])
BITFIELD(s_bytes, i_bit_array[])
BITFIELD(s_byte, string_out)
BITFIELD("This will work", i_bit_array[])
BITFIELD("\563F\", string_out)
```

Im nachfolgenden Beispiel ist die `sbyte`-Zeichenkette auf ein Hex-Byte eingestellt und wird zweimal an den `BITFIELD`-Befehl gesendet (einmal für ein Ganzzahl-Array und einmal für eine Zeichenkette).

```
COPY(sbyte: "\AE\")
BITFIELD(sbyte, ibits[])
BITFIELD(sbyte, sbits)
```

Aktueller Inhalt von Ausgabevariablen

```
ibits[0] = 1
ibits[1] = 0
ibits[2] = 1
ibits[3] = 0
ibits[4] = 1
ibits[5] = 1
ibits[6] = 1
ibits[7] = 0
sbits = "10101110"
```

BREAKPOINT



Mit dem `BREAKPOINT`-Befehl wird die Ausführung eines Parsing-Skripts angehalten. Wenn das Wizard-Fehlersuchprogramm für Skripts ausgeführt wird, stoppt der Breakpoint-Befehl den Parser (das Analyseprogramm) und wartet auf Benutzereingriff. Sie können im Panel des Wizard-Fehlersuchprogramms beispielsweise die Schaltfläche zum Starten bzw. für den Schritt aktivieren, um die Fehlersuche fortzusetzen.

Format

```
BREAKPOINT( )
```

BYTEFIELD



Der BYTEFIELD-Befehl akzeptiert die Byte einer Bit-Darstellung (0 oder 1) und nimmt sie in eine Zeichenkette auf.

Bei der Eingabe kann es sich um Folgendes handeln:

- Zeichenkette
- Ganzzahl-Array
- Float-Array

Bei der Ausgabe handelt es sich stets um eine Zeichenkettenvariable.

Format

ACHTUNG: Wenn es sich beim ersten Parameter um ein Ganzzahl- bzw. Float-Array handelt, verwenden Sie für `i_num_bytes` keine Werte größer als 100, da das Array in diese Anzahl von Einträgen initialisiert wird (wenn der Wert von `i_num_bytes` groß ist, kann sich dies als speicherintensiv erweisen).

`BYTEFIELD(source_var, s_bytes[, i_num_bytes])`

HINWEIS: Beim ersten Parameter für `BYTEFIELD` (`source_var`) muss es sich um `svar`, `ivar[]` oder `fvar[]` handeln.

Datentypen

Argument	Typ	Beschreibung
<code>source_var</code>	numvar-Array (EINGABE)	Array mit Ganzzahlen (auf 0 oder 1 eingestellt). Die Anzahl der Bit entspricht der Anzahl der Byte in <code>s_bytes</code> multipliziert mit 8. Für jeden 8-Bit-Satz erfolgt die Sortierung von Most Significant Bit (MSB) nach Least Significant Bit (LSB) (Beispiele unterhalb dieser Tabelle).
	svar (EINGABE)	Eine Zeichenkette, die ein Mehrfaches von 8 Byte enthält; hierbei steht jedes Byte für ein Bit in den eingegebenen Byte. Die Byte in dieser Zeichenkette sollten stets auf ASCII 0 oder 1 eingestellt werden. Für sämtliche aufeinanderfolgenden 8 Bit in den einzelnen Zeichenketten sollte die ASCII-Sortierung (0er und 1er) von MSB nach LSB erfolgen. Beispiel: Wenn <code>source_var = "0101101011111110"</code> und <code>i_num_bytes = 2</code> , Dann <code>s_bytes = "\5AFE\"</code>
<code>s_bytes</code>	Zeichenkette (AUSGABE)	Beliebige Anzahl von Byte mit ASCII- oder Hex-Daten in einer Zeichenkette.

Argument	Typ	Beschreibung
i_num_bytes	Numerisch (EINGABE) [OPTIONAL]	Die Anzahl der Byte, die in _bytes aufgenommen werden sollen. Da es sich um eine optionale Angabe handelt, ist 1 der Standard, es sei denn, die Verwendung erfolgt, wenn die Eingabe den Typ STRING aufweist. Wenn die Eingabe den Typ STRING aufweist, ist der Standard die Größe der Zeichenkette dividiert durch 8.

Spezifische Beispiele für source_var:

```

SOURCE_VAR[0] = MSB of Byte 1
SOURCE_VAR[1] = Next MSB of Byte 1
SOURCE_VAR[2] = Next MSB of Byte 1
SOURCE_VAR[3] = Next MSB of Byte 1
SOURCE_VAR[4] = Next MSB of Byte 1
SOURCE_VAR[5] = Next MSB of Byte 1
SOURCE_VAR[6] = Next MSB of Byte 1
SOURCE_VAR[7] = LSB of Byte 1
SOURCE_VAR[8] = MSB of Byte 2
SOURCE_VAR[9] = Next MSB of Byte 2
...
SOURCE_VAR[n * 8 - 1] = LSB of Byte n

```

Einige BYTEFIELD-Beispiele:

```

BYTEFIELD(i_bit_array[], s_bytes)
BYTEFIELD(string_bits_in, s_bytes)
BYTEFIELD(f_bit_array[], string_bytes, 2)
BYTEFIELD(i_bit_array[], string_bytes, i_num_bytes)

```

Im nachfolgenden Beispiel werden die sbyte-Zeichenkette und das ivar-Ganzzahl-Array auf eine Bit-Darstellung eines Hex-Byte eingestellt und zweimal an den BYTEFIELD-Befehl gesendet (einmal für die Ganzzahl-Array-Eingabe und einmal für die Zeichenketteneingabe).

```

SET(ivar[0] = 0)
SET(ivar[1] = 0)
SET(ivar[2] = 0)
SET(ivar[3] = 0)
SET(ivar[4] = 1)
SET(ivar[5] = 1)
SET(ivar[6] = 1)
SET(ivar[7] = 1)
COPY(sbits:"11110000")
BYTEFIELD(ivar[], sbyte1)
BYTEFIELD(sbits, sbyte2, 1)

```

Aktueller Inhalt von Ausgabevariablen

```
sbyte1 = "\0F\"
```

```
sbyte2 = "\F0\"
```

CLEAR



Mit dem CLEAR-Befehl werden Zeichenkettenvariablen auf null Byte gekürzt bzw. Ganzzahlvariablen und Float-Variablen auf null eingestellt. In einem einzelnen CLEAR-Befehl können bis zu 100 Variablen angegeben werden.

Format

```
CLEAR(<varlist>)
```

Hierbei gilt:

```
varlist ::= var [, <varlist>]
```

```
Var ::= variable to clear (fvar, ivar, or svar)
```

Maximale Anzahl an Variablen: 100

Datentypen

Argument	Typ	Beschreibung
var1	Variable (EINGABE/AUSGABE)	Die zu löschende Variable (fvar, ivar oder svar).
var2	Variable (EINGABE/AUSGABE) [OPTIONAL]	Die zu löschende Variable (fvar, ivar oder svar).
var3	Variable (EINGABE/AUSGABE) [OPTIONAL]	Die zu löschende Variable (fvar, ivar oder svar).
...	Variable (EINGABE/AUSGABE) [OPTIONAL]	Weitere zu löschende Variablen (fvar, ivar oder svar).

Beispiel:

```
CLEAR(var1)
```

```
CLEAR(var1,var2)
```

```
CLEAR(var1,var2,var3)
```

```
CLEAR(svar[45])
```

```
CLEAR(imatrix[5][5])
```

```
CLEAR(ivar, fvar, i_len, data_string[i_var])
```

```
CLEAR(temp)
```

```
CLEAR(sdata[index_x][index_y])
```

```
CLEAR(f_bits[3], i_var_array[2])
```

```
CLEAR(i_counter, temp)
```


In den nachfolgenden Beispielen werden Werte Zeichenkettenvariablen zugewiesen; die Zeichenkettenvariablen werden anschließend in einer Ereignismeldung verwendet und die Werte der jeweiligen Zeichenkettenvariable werden gelöscht.

```
COPY(res_var: "Firewall")
COPY(msg_var: "Firewall 116 Minor Alarm")
ALERT(res_var, msg_var, 4)
CLEAR(res_var, msg_var)
RESULT:
res_var = ""
msg_var = ""
```

CLEARTAGS



Mit dem CLEARTAGS-Befehl werden sämtliche reservierten Ereignis- und Datum-/Uhrzeit-Variablen gelöscht, die nicht durch den [CONSTANTTAGS](#)-Befehl geschützt werden.

Dieser Befehl sollte in der Initialisierungsphase (Phase 4 der Sentinel-Standardschablone) von Collector aufgerufen werden, bevor die Analysierung von Eingabe in die reservierten Variablen erfolgt.

Der CLEARTAGS-Befehl wird für die reservierten Ereignis- und Datum-/Uhrzeit-Variablen verwendet. Der CLEARTAGS-Befehl akzeptiert keine Parameter. Die Zeichenkettenvariablen sind auf die leere Zeichenkette ("") eingestellt. Beispiel:

```
s_EVT und s_Sec.
```

Die Ganzzahlvariable „i_Severity“ ist auf null eingestellt.

Format

```
CLEARTAGS ( )
```

Beispiel:

```
SET(i_Severity = 3)
COPY(s_BM: "Base Message")
COPY(s_Example: "Test")
CLEARTAGS()
```

Ergebnis:

```
i_Severity = 0
s_BM = ""
s_Example = "Test"
```

HINWEIS: „s_Example“ ist keine reservierte Ereignis- bzw. Datum-/Uhrzeit-Variable und wurde folglich nicht gelöscht.

COMMENT



Hierfür ist ein optionales Argument zulässig, bei dem es sich um eine Zeichenkette handelt. Dies ist eine Methode zur Eingabe von Kommentaren in die Collector-Schablonendatei. Auf diese Weise können Sie Kommentare vom visuellen Editor aus eingeben, ohne zum Texteditor wechseln zu müssen.

Format

```
/*[string]*/
```

Beispiel:

```
/* COLLECTOR INFORMATION
; -----
Collector_Name:           Standard Template
Collector_Description:    Template to base new Wizard
    Collectors on
Collector_Manufacturer:   N/A
Collector_Product/Version: N/A
Collector_Version:        release 4.1
Collector_Date:           August 2003
; -----*/
```

COMPARE



Der COMPARE-Befehl überprüft zwei Argumente und legt abhängig vom Ergebnis eine Variable fest. Das Ergebnis der Vergleichs des Zeichenkettentyps und des numerischen Typs kann in einer Variable gespeichert werden. Wenn es sich bei der Variable um eine ivar-Variable, eine fvar-Variable oder eine Zeichenkette handelt, enthält die Variable den Wert -1, 0 oder 1.

- -1 wird verwendet, wenn „arg1“ kleiner als „arg2“ ist
- 0 wird verwendet, wenn „arg1“ gleich „arg2“ ist
- 1 wird verwendet, wenn „arg1“ größer als „arg2“ ist

Format

```
COMPARE(arg1, arg2, dest)
```

Datentypen

Argument	Typ	Beschreibung
arg1	Beliebig (EINGABE)	Vergleichsdaten 1. Muss Zeichenkette oder numerisch sein.
arg2	Beliebig (EINGABE)	Vergleichsdaten 2. Muss denselben Typ wie Vergleichsdaten 1 aufweisen.
dest	Variable (AUSGABE)	Die Variable, in die das Ergebnis des Vergleichs geschrieben wird: svar = „-1“, „0“ oder „1“ ivar = -1, 0 oder 1 fvar = -1.0, 0.0 oder 1.0

HINWEIS: Die Typen von „arg1“ und „arg2“ müssen entweder beide numerisch sein bzw. es muss sich in beiden Fällen um eine Zeichenkette handeln.

Beispiel:

```
COMPARE(i_counter, 0, temp)
COMPARE(sdata, "ALM", i_sdata_cmp_val)
COMPARE(i_counter, i_counter2, temp)
COMPARE(i_counter, i_counter2, i_result[i_counter])
```

Im nachfolgenden Beispiel wird Text mit dem Inhalt einer Zeichenkettenvariablen verglichen und das Ergebnis des Vergleichs wird in einer Ganzzahlvariablen gespeichert. Wenn der Text nicht mit dem Wert der Zeichenkettenvariablen übereinstimmt, wird ein Ereignis erstellt.

```
COMPARE(s_data_var, "ALARM", i_compare_var)
IF(i_compare_var = 0)
  ALERT(res_var, "Major ALARM", 5)
ENDIF( )
```

HINWEIS: Die IF()-,ELSE()- und ENDIF()-Befehle führen dieselbe Funktion wie der COMPARE-Befehl aus, der einzige Unterschied besteht im Vergleich negativer Zahlen.

CONSTANTTAGS



Für den CONSTANTTAGS-Befehl ist eine variable Anzahl an Parametern reservierter Variablennamen (Ereignis und Datum/Uhrzeit) zulässig. Durch das Deklarieren einer reservierten Variablen als konstant sorgt er dafür, dass die Variable durch den Aufruf des [CLEARTAGS](#)-Befehls nicht gelöscht wird.

„s_PN“ ist ein Beispiel für eine Variable dieser Art; hier ist der Produktname enthalten, den der Collector verarbeitet. Die s_PN-Variable sollte als konstant deklariert und einmalig in der Collector-Einrichtungsphase festgelegt werden.

Dieser Befehl sollte in der Collector-Einrichtungsphase (Phase 1 in der 4.1-Standardschablone) für reservierte Variablen aufgerufen werden, die bei der Verarbeitung von Ereignissen durch den Collector unverändert bleiben.

Der [CONSTANTTAGS](#)-Befehl wird für die reservierten Ereignis- und Datum-/Uhrzeit-Variablen verwendet.

Format

```
CONSTANTTAGS (<reserved_variable> [, ...])
```

Datentypen

Argument	Typ	Beschreibung
reserved_variable		Die Liste der reservierten Variablen, die als konstant festgelegt werden und durch den CLEARTAGS-Befehl nicht gelöscht werden.

Beispiel:

```
COPY(s_PN: "PN" )
COPY(s_ST: "ST" )
COPY(s_BM: "BM" )
CONSTANTTAGS(s_PN, s_ST)
CLEARTAGS( )
```

Ergebnis:

```
s_PN = "PN"
s_ST = "ST"
s_BM = ""
```

Von den drei reservierten Ereignisvariablen wurde s_BM nicht durch [CONSTANTTAGS](#) vor [CLEARTAGS](#) geschützt und folglich gelöscht.

CONVERT



Mit dem CONVERT-Befehl wird eine Eingabezeichenkette (Binär-, Oktal-, Dezimal-, Hex- bzw. Raw-Zeichenkette) in eine Ausgabezeichenkette (Binär-, Oktal-, Dezimal-, Hex- bzw. Raw-Zeichenkette) umgewandelt.

Format

```
CONVERT(string_in, type_in, svar_out, type_out)
```

Datentypen

Argument	Typ	Beschreibung
string_in	Zeichenkette (EINGABE)	Die umzuwandelnde Eingabezeichenkette.
type_in	Auswahlliste Zeichenkette Zeichenkettenvariable (EINGABE)	Der Typ der Eingabezeichenkette. (string_in): Binär = „B“ oder „b“ Oktal = „O“ oder „o“ Dezimal = „D“ oder „d“ Hex = „H“ oder „h“ Raw = „R“ oder „r“
svar_out	svar (AUSGABE)	Die Zeichenkettenvariable, die die umgewandelten Zeichenkettendaten enthält.

Argument	Typ	Beschreibung
type_out	Auswahlliste Zeichenkette Zeichenkettenvariable (EINGABE)	Der Typ, in den die Daten umgewandelt werden sollen (umgewandelte Zeichenkette wird in svar_out gespeichert): Binär = „B“ oder „b“ Oktal = „O“ oder „o“ Dezimal = „D“ oder „d“ Hex = „H“ oder „h“ Raw = „R“ oder „r“

Beispiel:

```

CONVERT("10101010", "b", shex, "h")
CONVERT(sdata, "B", sraw, "r")
CONVERT("2356", "d", soctal, "o")
CONVERT("\3A\ ", "r", sbinary, "b")
CONVERT("2A3E", "h", sraw, "r")
CONVERT(data, "r", sdecimal, "d")
CONVERT(data, "o", shex, "H")

```

Im nachfolgenden Beispiel wird der CONVERT-Befehl für mehrere Umwandlungen aufgerufen.

```

CONVERT("\0afe\ ", "R", sdecimal, "D")
CONVERT("63", "d", sbinary, "b")
CONVERT("63", "d", shex, "h")
CONVERT("63", "d", soctal, "o")
CONVERT("1101010111110101", "b", sraw, "r")

```

Aktueller Inhalt von Ausgabevariablen:

```

sdecimal = "2814"
sbinary = "00111111"
shex = "3F"
soctal = "077"
sraw = "\d5 f5\ "

```

COPY



Mit dem COPY-Befehl werden Daten aus dem Empfangspuffer bzw. der Quellzeichenkette dupliziert und in eine Zeichenkettenvariable bzw. eine Zeichenkette mit Anführungszeichen für eine Zeichenkette geschrieben. Der Rx Buffer Pointer bleibt bei Verwendung dieses Befehls unverändert.

Das Ziel für die Daten (svar) muss mithilfe der COPY-Parameter angegeben werden.

HINWEIS: Im visuellen Editor von Collector Builder sind COPY, COPY-FROM-RX-BUFF-UNTIL-SEARCH, COPY-FROM-RX-BUFF, COPY-FROM-STRING-TO-STRING-UNTIL-SEARCH und COPY-STRING-TO-STRING als separate Befehle aufgelistet. Es handelt sich hierbei um ein- und denselben Befehl. Sie werden als Beschreibungen von Varianten desselben Befehls zur Verfügung gestellt. Wenn Sie im Texteditor eine der Varianten des COPY-Befehls verwenden möchten, geben Sie COPY ein.

Bei der Verwendung dieses Befehls gilt Folgendes:

- Es muss ein Offset in die Quelle angegeben, um zu steuern, wohin Daten aus den Quelldaten kopiert werden.
- Die Anzahl der Byte, die in die Zielvariable kopiert werden sollen, kann mithilfe des Längenparameters (ilen) angegeben werden; anderenfalls kann standardmäßig die Länge der Quelldaten verwendet werden.
- Zusätzlich zur Angabe eines numerischen Längenparameters kann eine Zeichenkette verwendet werden. Durch die Verwendung einer Zeichenkette kopiert die Collector-Engine Byte aus den Quelldaten (beginnend bei Offset) an die Zielvariable an, und zwar bis zum, aber ohne das erste Zeichen der Zeichenkette (wenn sie gefunden wird). Wenn die Zeichenkette nicht gefunden wird, werden keine Byte kopiert.
- Wenn der Offsetparameter (ioffset) bzw. der Längenparameter (ilen) außerhalb des Bereichs der Quellvariable liegen, werden so viele Byte wie möglich, bis zum Ende der Quelldaten, kopiert.

Wenn der Offset der Länge der Quelldaten entspricht bzw. diese übersteigt, werden keine Byte in die Zielvariable kopiert.

Wenn kein Offset angegeben wird, wird standardmäßig Null verwendet.

Format

```
COPY(<DEST>: [SOURCE] [, [SEARCH] [, [ILEN] [, [IOFFSET]
    ] ] ] )
COPY(<DEST>: [SOURCE] [, [ILEN] [, [IOFFSET] ] ] )
COPY(<DEST>: [ILEN] [, [OFFSET] ] ] )
```

Datentypen

Argument	Typ	Beschreibung
dest	svar (AUSGABE)	Die Datenzeichenkettenvariable, in die Byte kopiert werden.
source string	(EINGABE) [OPTIONAL] Oder svar	Die Zeichenkette, aus der Byte kopiert werden (Standard = Empfangspuffer). Wenn der Suchparameter verwendet wird.
search	Zeichenkette (EINGABE) [OPTIONAL]	Eine Zeichenkette, mit der Folgendes angegeben wird: Kopieren bis zu den Byte, nach denen in der Quellzeichenkette gesucht werden soll.
ilen	Numerisch (EINGABE) [OPTIONAL]	Die Anzahl der Byte, die von der Quelle in das Ziel kopiert werden sollen.
ioffset	Numerisch (EINGABE) [OPTIONAL]	Der Offset in der Quelle, bei dem mit dem Kopieren von Daten begonnen werden soll; kopiert alle Zeichen aus dem Empfangspuffer in den Übertragungspuffer.

In den nachfolgenden Beispielen werden Byte vom Empfangspuffer in eine Ziel-svar (dest) kopiert. Die Rx Buffer Pointer Position wird dem Offset-Wert hinzugefügt, um die erste Position der Daten für das Kopieren anzugeben. Das ^-Symbol kennzeichnet die Rx Buffer Pointer Position.

Es gelten folgende Annahmen:

```

rxbuff="receive buffer"
^ (Rx buffer pointer position)
dest=""
source="A source string"
ilen=3
ioffset=3

```

Befehl	Ergebnis
COPY(dest:)	dest = "receive buffer"
COPY(dest:5)	dest = "recei"
COPY(dest:,5)	dest = "ve buffer"

In den nachfolgenden Beispielen werden Byte von einer Quellzeichenkette in eine Ziel-svar (dest) kopiert.

Befehl	Ergebnis
COPY(dest:source)	dest = "A source string"
COPY(dest:source,5)	dest = "A sou"
COPY(dest:source,5,6)	dest = "ce st"

In den nachfolgenden Beispielen werden Byte aus dem Empfangspuffer in eine Zeichenkettenvariable kopiert, und zwar bis zur, jedoch ohne die Suchzeichenkette. Wenn im Empfangspuffer keine Suchzeichenkette gefunden wird (nach Rx Buffer Pointer + Offset Position), werden keine Byte kopiert.

HINWEIS: Bei der Hex-Ersetzung wird mit \0000\ eine Zeichenkette beendet. Folglich wird aus „xxxx\0000\yyyy“ „xxxx“.

In den nachfolgenden Beispielen werden Byte vom Empfangspuffer in eine Ziel-svar (dest) kopiert, und zwar bis zur, jedoch ohne die Suchzeichenkette. Wenn im Empfangspuffer keine Suchzeichenkette gefunden wird (nach Rx Buffer Pointer + Offset Position), werden keine Byte kopiert.

Befehl	Ergebnis
<code>COPY(dest:,"buffer")</code>	<code>dest = "receive "</code>
<code>COPY(dest:,"receive")</code>	<code>dest = ""</code>

In den nachfolgenden Beispielen werden Byte aus einer Quellzeichenkette (muss eine Zeichenkettenvariable sein) in eine Zielzeichenkettenvariable (dest) kopiert, und zwar bis zur, jedoch ohne die Suchzeichenkette. Wenn im Empfangspuffer keine Suchzeichenkette gefunden wird (nach Rx Buffer Pointer + Offset Position), werden keine Byte kopiert.

Befehl	Ergebnis
<code>COPY(dest:source," string")</code>	<code>dest = "a source"</code>
<code>COPY(dest:source," .string")</code>	<code>dest = ""</code>

CRC



Mit dem CRC-Befehl wird eine zyklische Redundanzprüfung für eine Zeichenkette von Byte (Hex oder ASCII) berechnet).

Format

```
CRC(source_data, dest_crc)
```

Datentyp

Argument	Typ	Beschreibung
source_data	Zeichenkette (EINGABE)	Die Zeichenkettenvariablen, für die der CRC-Befehl ausgeführt werden soll.
dest_crc	svar (AUSGABE)	Die Zeichenkettenvariable, in der das 2 Byte umfassende CRC-Ergebnis gespeichert wird.

Beispiel:

Im nachfolgenden Beispiel wird der berechnete CRC-Wert mit einem gespeicherten Wert verglichen. Wenn die beiden CRC-Werte identisch sind, wird eine Ereignismeldung ausgegeben.

```
CRC(svar, s_crc_var)
IF(s_crc_var = "\0A5F\")
EVENT(res, "Correct CRC generated", 0)
ENDIF( )
```

HINWEIS: Bei der Hex-Ersetzung wird mit \0000\ eine Zeichenkette beendet; folglich wird aus „xxxx\0000\yyyy“ „xxxx“.

DATE



Mit dem DATE-Befehl wird das aktuelle Datum (im Format MM-TT-JJJJ) in eine Zeichenkettenvariable kopiert. Optional kann damit der aktuelle Wochentag in eine Zeichenkette bzw. eine Ganzzahl- oder Float-Variable kopiert werden.

Format

```
DATE(date_string [, day_of_week] [, i_day_of_week]
    [, f_day_of_week])
```

Datentyp

Argument	Typ	Beschreibung
date_string	svar (AUSGABE)	Die Zeichenkettenvariable, in der das Datum gespeichert wird (Beispiel: svar = „11-18-2002“).
day_of_week	svar (AUSGABE) [OPTIONAL] ivar (AUSGABE) [OPTIONAL] Oder fvar (AUSGABE) [OPTIONAL]	(Optional) Die Zeichenkettenvariable, in der der Wochentag gespeichert wird; die Angabe erfolgt als vollständig ausgeschriebene Bezeichnung des Tages (Beispiel: svar = Saturday) (Optional) Die Zeichenkettenvariable, in der der Wochentag gespeichert wird; die Angabe erfolgt als vollständig ausgeschriebene Bezeichnung des Tages = Zahl: Monday = 1 Tuesday = 2 Wednesday = 3 Thursday = 4 Friday = 5 Saturday = 6 Sunday = 7 (Beispiel: Monday ist ivar = 1)

Beispiel:

Im nachfolgenden Beispiel wird das Datum auf dem System mit einer Datumszeichenkette verglichen. Wenn die beiden Datumsangaben identisch sind, wird eine Ereignismeldung ausgegeben.

```
DATE(date_var, day_of_week)
IF(date_var = "11-18-2002")
ALERT(res, "Happy 23rd birthday!", 0)
ENDIF( )
IF(day_of_week = "Saturday")
ALERT(res, "Time to go to the beach," 0)
ENDIF( )
```

DATETIME



Mit dem DATETIME-Befehl wird eine Ganzzahldarstellung der Anzahl der Sekunden seit dem 1. Januar 1970 in Datums- und Uhrzeit-Zeichenkettenvariablen umgewandelt. Optional kann damit der aktuelle Wochentag in eine Zeichenkette bzw. eine Ganzzahl- oder Float-Variable kopiert werden.

Format

```
DATETIME(itime_secs, svar_date, svar_time [, day_of_week]
        [, i_day_of_week] [, f_day_of_week])
```

Datentypen

Argument	Typ	Beschreibung
itime_secs	Numerisch (EINGABE)	Die Ganzzahl, die die Anzahl der Sekunden seit 1970 enthält.
svar_date	svar (AUSGABE)	Die Zeichenkettenvariable, in der das Datum gespeichert wird (Beispiel: 02-19-96).
svar_time	svar (AUSGABE)	Die Zeichenkettenvariable, in der die Uhrzeit gespeichert wird (Beispiel: 15:14:33).
day_of_week	svar (AUSGABE) [OPTIONAL] ivar (AUSGABE) [OPTIONAL] Oder fvar (AUSGABE) [OPTIONAL]	(Optional) Die Zeichenkettenvariable, in der der Wochentag gespeichert wird; die Angabe erfolgt als vollständig ausgeschriebene Bezeichnung des Tages (Beispiel: svar = Saturday) (Optional) Die Zeichenkettenvariable, in der der Wochentag gespeichert wird; die Angabe erfolgt als vollständig ausgeschriebene Bezeichnung des Tages = Zahl: Monday = 1 Tuesday = 2 Wednesday = 3 Thursday = 4 Friday = 5 Saturday = 6 Sunday = 7 (Beispiel: Monday ist ivar = 1)

Beispiel:

Im nachfolgenden Beispiel wird mit dem DATETIME-Befehl die Anzahl der Sekunden seit 1970 in Datums- und Uhrzeitzeichenketten umgewandelt:

```
DATETIME(0, sdatevar, stimevar)
```

Im nachfolgenden Beispiel werden mit dem DATETIME-Befehl der Wochentag sowie Datum und Uhrzeit angegeben:

```
DATETIME(946728000, sdate, stime, sday)
```

Aktueller Inhalt von Ausgabevariablen:

```
sdatevar = "01-01-70"  
stimevar = "00:00:00"  
sdate = "01-01-2000"  
stime = "12:00:00"  
sday = "Saturday"
```

DBCLOSE



Mit dem DBCLOSE-Befehl wird die Datenbankverbindung geschlossen. Zwei Parameter sind erforderlich.

- Beim ersten erforderlichen Parameter handelt es sich um die Datenbankzugriffsnummer, die vom [DBOPEN](#)-Befehl zurückgegeben wird. Hierbei handelt es sich entweder um eine Ganzzahl oder eine Ganzzahlvariable.
- Beim zweiten erforderlichen Parameter handelt es sich um den Status des Schließvorgangs. Hierbei handelt es sich entweder um eine Ganzzahl- oder eine Float-Variable. Bei Erfolg wird „1“ zurückgegeben.

Format

```
DBCLOSE(i_dbhandle, i_closestatus)
```

DBDELETE



Mit dem DBDELETE-Befehl werden basierend auf Auswahlkriterien Zeilen aus der ausgewählten Tabelle gelöscht. Vier Parameter sind erforderlich.

- Beim ersten erforderlichen Parameter handelt es sich um die Datenbankzugriffsnummer, die vom [DBOPEN](#)-Befehl zurückgegeben wird. Hierbei handelt es sich entweder um eine Ganzzahl oder eine Ganzzahlvariable.
- Beim zweiten erforderlichen Parameter handelt es sich um den Status des Löschvorgangs. Hierbei handelt es sich entweder um eine Ganzzahl- oder eine Float-Variable. Die Anzahl der gelöschten Zeilen werden bei Erfolg zurückgegeben (einschließlich 0).
- Beim dritten erforderlichen Parameter handelt es sich um den Namen der Tabelle, aus der Zeilen gelöscht werden sollen. Hierbei kann es sich entweder um eine Zeichenkette oder eine Zeichenkettenvariable handeln.
- Beim vierten (optionalen) Parameter handelt es sich um die where-Klausel. Mit ihrer Hilfe können Benutzer nicht benötigte Daten durch ein Auswahlkriterium herausfiltern. Wenn hier keine Angabe erfolgt, werden beim Löschvorgang sämtliche Zeilen der Tabelle gelöscht.

Die Fehlercodes für den DBDELETE-Befehl lauten wie folgt:

```
>0No error  
0No rows deleted  
-1DB handle is invalid
```

Format

```
DBDELETE(i_dbhandle, i_deletestatus, "tablename", "where  
clause")
```

Beispiel:

```
DBDELETE(i_dbhandle, i_deletestatus, "tablename")  
DBDELETE(i_dbhandle, i_deletestatus, s_tablename, "where  
clause")
```

DBGETROW



Der DBGETROW-Befehl wird zusammen mit dem [DBSELECT](#)-Befehl verwendet. Der Benutzer muss zunächst eine Auswahl tätigen (mit [DBSELECT](#)), bevor mit dem DBGETROW-Befehl Zeilen abgerufen werden können. Mit diesem Befehl wird die nächste verfügbare Zeile in einer Auswahl abgerufen; hierbei wird ein Cursor offen gelassen, damit dieser Befehl als Schleife aufgerufen und so beim jeweiligen nächsten Aufruf die nächste Zeile abgerufen werden kann. Vier Parameter sind erforderlich.

- Beim ersten erforderlichen Parameter handelt es sich um die Datenbankzugriffsnummer, die vom [DBOPEN](#)-Befehl zurückgegeben wird. Hierbei kann es sich entweder um eine Ganzzahl oder eine Ganzzahlvariable handeln.
- Beim zweiten erforderlichen Parameter handelt es sich um die Zugriffsnummer für den Auswahlvorgang. Hierbei kann es sich entweder um eine Zeichenkette oder eine Zeichenkettenvariable handeln. Hierbei handelt es sich um dieselbe Zugriffsnummer, die auch bei der Ausführung des [DBSELECT](#)-Befehls zugewiesen wurde.
- Beim dritten erforderlichen Parameter handelt es sich um den Status des Abrufvorgangs. Hierbei handelt es sich entweder um eine Ganzzahl- oder eine Float-Variable. Bei Erfolg wird „1“ zurückgegeben.
- Beim vierten erforderlichen und nachfolgend optionalen Parameter handelt es sich um die vom Befehl zurückgegebenen Spaltendaten. Bei diesen Spalten kann es sich um Zeichenkettenvariablen, Float-Variablen oder Ganzzahlvariablen handeln. Spaltendaten, die nicht mit dem Parametertyp übereinstimmen, werden, wenn möglich, in den passenden Parametertyp umgewandelt. Wenn die Tabelle also eine Float-Spalte enthält, es sich beim Parameter jedoch um eine Zeichenkette handelt, werden die Float-Daten in Zeichenkettendaten umgewandelt. Der Benutzer kann bis zu 48 dieser Parameter aufnehmen.

HINWEIS: Mit dem Befehl wird die geringere Anzahl der definierten Parameter sowie der tatsächlich in der Datenbank vorhandenen Spalten gefüllt. Wenn die Datenbank 4 Spalten aufweist, Sie jedoch 7 dieser Parameter angeben, werden nur die ersten 4 gefüllt.

Die Fehlercodes für den DBGETROW-Befehl lauten wie folgt:

```
1No Error  
-1Error retrieving row
```

Format

```
DBGETROW(i_dbhandle, "select1", i_selectstatus, s_col1,
        s_col2, s_col3, ..., s_col48)
```

Beispiel:

```
DBGETROW(i_dbhandle, s_selecthandle, i_selectstatus,
        s_col1, s_col2)
```

DBINSERT



Mit dem DBINSERT-Befehl wird für eine ausgewählte Tabelle eine Zeile mit Daten in die Datenbank eingefügt. Vier Parameter sind erforderlich.

- Beim ersten erforderlichen Parameter handelt es sich um die Datenbankzugriffsnummer, die vom [DBOPEN](#)-Befehl zurückgegeben wird. Hierbei handelt es sich entweder um eine Ganzzahl oder eine Ganzzahlvariable.
- Beim zweiten erforderlichen Parameter handelt es sich um den Status des Einfügevorgangs. Hierbei handelt es sich entweder um eine Ganzzahl- oder eine Float-Variable. Bei Erfolg wird „1“ zurückgegeben.
- Beim dritten Parameter handelt es sich um den Namen der Tabelle, in die die Daten eingefügt werden sollen.
- Beim vierten erforderlichen und nachfolgend optionalen Parameter handelt es sich um die einzufügenden Spaltendaten. Diese Spalten können einen beliebigen Typ aufweisen. Der Benutzer kann bis zu 48 dieser Parameter aufnehmen.

Der Befehl muss die genaue Anzahl der Parameter enthalten, die zum Einfügen einer Zeile mit Daten erforderlich sind. DBINSERT fügt keinen neuen Datensatz hinzu, wenn gegen eine eindeutige Beschränkung verstoßen wird.

Die Fehlercodes für den DBINSERT-Befehl lauten wie folgt:

```
1 No Error
-1 DB Handle is invalid / no row inserted
-2 Data request cannot be created
-7 SQL execution error
-16 SQL syntax error
```

Format

```
DBINSERT(i_dbhandle, i_insertstatus, "theTableName",
        "data1", "data2", ..., "data48")
```

Beispiel:

```
DBINSERT(i_dbhandle, i_insertstatus, s_theTableName,
        "data1", I_data2, f_data3)
DBINSERT(i_dbhandle, i_insertstatus, "theTableName",
        s_data1, "data2")
```

DBOPEN



Mit dem DBOpen-Befehl wird eine Verbindung zu einer unterstützten Datenbank geöffnet.

Nur im Microsoft Windows NT Collector funktioniert DBOpen nicht, wenn der Datenbankname so konfiguriert wurde, dass er auf ein „zugeordnetes Laufwerk“ verweist. Da der Collector als Service ausgeführt wird, wird er (normalerweise) über das „System“-Konto ausgeführt. Dieses Konto ist nicht berechtigt, auf Remote-Freigaben, einschließlich zugeordneter Laufwerke, zuzugreifen. Jede Datenbankverbindung (auch über ODBC (On Board Diagnostic Computer)) in einem Windows-Collector muss mit einer rein lokalen Datenbank erfolgen.

Fünf Parameter sind erforderlich.

- Beim ersten erforderlichen Parameter handelt es sich um den Datenbanktyp. Er kann über eine Auswahlliste bzw. mithilfe einer Zeichenkette bzw. Zeichenkettenvariable ausgewählt werden. „Oracle9i“ ist der zulässige Wert für diesen Parameter.
- Beim zweiten erforderlichen Parameter handelt es sich um den Namen der Datenbank, mit der die Verbindung hergestellt werden soll. Hierbei kann es sich entweder um eine Zeichenkette oder eine Zeichenkettenvariable handeln.
- Beim dritten erforderlichen Parameter handelt es sich um den Benutzernamen für die Datenbank. Hierbei kann es sich entweder um eine Zeichenkette oder eine Zeichenkettenvariable handeln. Dieses Feld kann beliebigen Text enthalten, wenn die Benutzereinrichtung nicht ausdrücklich für den Zugriff auf die Datenbank erfolgt ist.
- Beim vierten erforderlichen Parameter handelt es sich um das Passwort für den Benutzer. Hierbei kann es sich entweder um eine Zeichenkette oder eine Zeichenkettenvariable handeln. Dieses Feld kann beliebigen Text enthalten, wenn die Benutzereinrichtung nicht ausdrücklich für den Zugriff auf die Datenbank erfolgt ist.
- Beim fünften erforderlichen Parameter handelt es sich um die Datenbankzugriffsnummer, die von diesem Befehl in der Ganzzahlvariable bzw. Float-Variable zurückgegeben wird. Bei Erfolg ist die Datenbankzugriffsnummer größer als 0.

Format

```
DBOPEN("oracle9i", "Database name", "username",  
      "password", i_dbhandle)
```

Beispiel:

```
DBOPEN(s_dbtype, s_dbname, s_username, s_password,  
      i_dbhandle)  
DBOPEN(s_dbtype, "dbname", s_username, "password",  
      i_dbhandle)
```

DBSELECT



Der DBSELECT-Befehl wird zusammen mit dem DBGETROW-Befehl verwendet. Mit dem DBSELECT-Befehl wird ein Auswahlcursor für die Datenbank geöffnet. Dieser ruft einen Snapshot der aktuellen Datensätze in der Datenbank ab, die den Auswahlkriterien entsprechen. Nach dem DBSELECT-Befehl eingegebene Datensätze werden beim Datensatzabruf erst angezeigt, wenn die Auswahl durch Ausgabe eines weiteren DBSELECT-Befehls aktualisiert wird.

Sieben Parameter sind erforderlich.

- Beim ersten erforderlichen Parameter handelt es sich um die Datenbankzugriffsnummer, die vom [DBOPEN](#)-Befehl zurückgegeben wird. Hierbei handelt es sich entweder um eine Ganzzahl oder eine Ganzzahlvariable.
- Beim zweiten erforderlichen Parameter handelt es sich um den Status des Auswahlvorgangs. Hierbei handelt es sich entweder um eine Ganzzahl- oder eine Float-Variable. Bei Erfolg wird „1“ zurückgegeben.
- Beim dritten erforderlichen Parameter handelt es sich um die Auswahlkennung. Hierbei kann es sich entweder um eine Zeichenkette oder eine Zeichenkettenvariable handeln. Bei mehreren DBSELECT-Befehlen sollte sie eindeutig sein.
- Beim vierten erforderlichen Parameter handelt es sich um die Anzahl der Zeilen, die nach Abschluss des Auswahlvorgangs übersprungen werden sollen. Auf diese Weise kann der Benutzer den Zeiger im [DBGETROW](#)-Befehl für neue Daten positionieren und alte Daten können gleichzeitig übersprungen werden. Hierbei kann es sich entweder um eine Ganzzahl oder eine Ganzzahlvariable handeln.
- Beim fünften erforderlichen Parameter handelt es sich um die Tabelle, aus der die Daten abgerufen werden sollen. Hierbei kann es sich entweder um eine Zeichenkette oder eine Zeichenkettenvariable handeln.
- Beim sechsten (optionalen) Parameter handelt es sich um die where-Klausel. Mit ihrer Hilfe können Benutzer nicht benötigte Daten durch ein Auswahlkriterium herausfiltern. Wenn hier keine Angabe erfolgt, umfasst der Auswahlvorgang sämtliche Zeilen der Tabelle. Format der where-Klausel: where column-name='data'.
- Beim siebten (optionalen) Parameter handelt es sich um die vom DBSELECT-Befehl zurückgegebenen Spalten. Wenn hier keine Angabe erfolgt, umfasst der Auswahlvorgang sämtliche Spalten der Tabelle.

Die Fehlercodes für den DBSELECT-Befehl lauten wie folgt:

```
1 No Error
-1 DB_Handle is invalid
-2 Data request cannot be created
-3 Unsuccessful autocommit setting
-4 Memory allocation error
-5 SQL syntax error
-6 SQL execution error
```

Format

```
DBSELECT( i_dbhandle, i_selectstatus, "select1",  
          i_rows_to_skip, "f_atom"<, "where clause"><,  
          "coll1<col2><...>">)
```

Beispiel:

```
DBSELECT(i_dbhandle, i_selectstatus, "select1",  
          i_rows_to_skip, "f_atom")  
DBSELECT(i_dbhandle, i_selectstatus, s_select1, 23,  
          S_TABLENAME, s_whereclause)  
DBSELECT(i_dbhandle, i_selectstatus, s_select1, 23,  
          S_TABLENAME, "where fname='BOB'")  
DBSELECT(i_dbhandle, i_selectstatus, s_select1, 23,  
          S_TABLENAME, "where fname='BOB'", "FIRST, LAST,  
          ADDRESS")
```

DEC



Mit dem DEC-Befehl wird eine numerische Variable um 1 verringert. Bei der Verwendung von DEC muss entweder eine ivar oder eine fvar angegeben werden.

Format

```
DEC(i_numvar)
```

Datentypen

Argument	Typ	Beschreibung
i_numvar	numvar (EINGABE/AUSGABE)	Die zu verringernde Variable (ivar oder fvar)

Beispiel:

```
SET(icounter = 2)  
DEC(icounter)  
DEC(icounter)
```

Ergebnis:

```
icounter = 0
```


DECODE



Mit dem DECODE-Befehl wird eine Zeichenkette zurückgewandelt, die zur Aufrechterhaltung der Paketidentifizierung verschlüsselt wurde. Dieser Befehl identifiziert die Match-Byte (oder -Zeichen) und die Escape-Byte (oder -Zeichen), um das Escape-Zeichen zu entfernen. Er entfernt jedes Vorkommen der Escape-Zeichenkette vor den abgeglichenen Byte (Matched Bytes) in den Daten.

Format

```
DECODE(data_decode, match, escape)
```

Datentypen

Argument	Typ	Beschreibung
data_decode	svar (EINGABE/AUSGABE)	Die zu entschlüsselnde Zeichenkettendatenvariable. Das entschlüsselte Ergebnis wird wieder in diese Variable geschrieben.
match	Zeichenkette (EINGABE)	Die Zeichenkette von Byte zur Abstimmung in der data_decode-Zeichenkettenvariablen.
escape	Zeichenkette (EINGABE)	Die Escape-Zeichenkette, die aus der data_decode-Variable entfernt werden soll.

Beispiel:

Im nachfolgenden Beispiel wird eine Zeichenkette verschlüsselt, zum Speichern der verschlüsselten Version kopiert und anschließend mit denselben Parametern entschlüsselt.

```
COPY(svar:"This is just a test of decode")
ENCODE(svar, " ", "\00\")
COPY(svar_encode:svar)
DECODE(svar, " ", "\00\")
```

Aktueller Inhalt von Ausgabevariablen:

```
svar = "This is just a test of decode"
svar_encode = "This\00\ is\00\ just\00\ a\00\ test\00\
of\00\ decode"
```

DECODEMIME



Mit dem DECODEMIME-Befehl kann der Benutzer eine mit Base 64 verschlüsselte Zeichenkette bzw. Zeichenkettenvariable mithilfe der Base 64-Entschlüsselung entschlüsseln und die resultierende entschlüsselte Zeichenkette in einer Zeichenkettenvariablen speichern. Wenn ein Fehler auftritt, weist die resultierende Datenzeichenkette eine Länge von null auf und der Erfolgswert für die optionale Zahlenvariable wird auf „0“ eingestellt. Wenn die Entschlüsselung erfolgreich war, wird der Erfolgswert der Zahlenvariable auf „1“ eingestellt.

Format

```
DECODEMIME(encoded_data, data, success)
```

Datentypen

Argument	Typ	Beschreibung
encoded_data	Zeichenkette/Zeichenkettenvariable (EINGABE)	Mit Base 64 verschlüsselte Zeichenkette, die entschlüsselt werden muss.
data	Zeichenkettenvariable (AUSGABE)	Resultierende entschlüsselte Daten.
success	Ganzzahlvariable/Float-Variable (AUSGABE) [OPTIONAL]	Bei erfolgreicher Entschlüsselung auf „1“ eingestellt, bei einem Fehler auf „0“.

Beispiel:

```
DECODEMIME("VGVzdGluZyBEYXRhIEVuY29kaW5n", s_data,  
            i_success)
```

Im obigen Beispiel wird mit dem DECODEMIME-Befehl die Zeichenkette in doppelten Anführungszeichen mithilfe der 64 Base-Entschlüsselung entschlüsselt und die resultierende entschlüsselte Zeichenkette wird in s_data gespeichert. S_data wird folgendermaßen gefüllt:

```
test encode64 command
```

Da die Entschlüsselung erfolgreich war, wird der i_success-Ganzzahlvariable der Wert „1“ zugewiesen.

Ziehen Sie auch den [ENCODEMIME](#)-Befehl zurate.

DELETE



Mit dem DELETE-Befehl werden Variablen vom System entfernt, um für deren Speicherung vorgesehenen Arbeitsspeicher freizugeben (besonders nützlich bei Zeichenkettenvariablen).

Es empfiehlt sich, svars zu löschen, wenn Sie die entsprechenden Vorgänge abzuschließen, um so wenig Arbeitsspeicher wie möglich zu beanspruchen. In einem einzelnen DELETE-Befehl können bis zu 100 Variablen angegeben werden.

Format

```
DELETE(<varlist>)
```

Hierbei gilt:

```
varlist ::= var [, <varlist>]  
Var ::= variable to clear (fvar, ivar, or svar)
```

Maximale Anzahl an Variablen: 100

Datentypen

Argument	Typ	Beschreibung
var1	Variable (EINGABE/AUSGABE)	Die zu löschende Variable (fvar, ivar oder svar).
var2	Variable (EINGABE/AUSGABE) [OPTIONAL]	Die zu löschende Variable (fvar, ivar oder svar).
var3	Variable (EINGABE/AUSGABE) [OPTIONAL]	Die zu löschende Variable (fvar, ivar oder svar).
...	Variable (EINGABE/AUSGABE) [OPTIONAL]	Weitere zu löschende Variablen (fvar, ivar oder svar).

Beispiel:

```
DELETE(ivar1)
DELETE(sdata, i_len, i_count, svar[22])
DELETE(imatrix3d[ix][iy][iz])
DELETE(f_array[i_count], svar[4], sdata)
DELETE(ichart[3][icount])
```

DISPLAY



Mit dem DISPLAY-Befehl werden die Skriptvariablen und ihre aktuellen Werte in einem Popup-Fenster angezeigt.

Sie können folgende Schritte durchführen:

- Diesen Befehl bei der Fehlersuche in Skripten verwenden
- Wenn Sie eine Zeichenkette als Parameter übergeben, wird der Inhalt dieser Zeichenkette angezeigt
- Zeichenketten, die Hex-Daten enthalten, werden im Hex-Format angezeigt (also string="\0a 0d")

Das Programm versucht zunächst, die Zeichenkette in ASCII (American Standard Code for Information Interchange) anzuzeigen. Wenn die Zeichenkette sowohl druckbare als auch nicht druckbare Hex-Daten enthält, werden die druckbaren Hex-Zeichen im ASCII- und der Rest der Zeichenkette im Hex-Format angezeigt. Bei der Hex-Ersetzung wird mit \0000\ eine Zeichenkette beendet; folglich wird aus „xxxx\0000\yyyy“ „xxxx“.

Format

```
DISPLAY(string_data)
```

Datentypen

Argument	Typ	Beschreibung
string_data	Zeichenkette	Eine anzuzeigende Zeichenkette.
	(EINGABE) [OPTIONAL]	Wenn dies nicht ausgewählt wird, wird für alle Skripts sämtlicher Variableninhalt (Zeichenketten, Zahlen und Arrays) angezeigt.

Beispiel:

```
DISPLAY( )  
DISPLAY(sdata_var)  
DISPLAY("Hello This is String Data")  
DISPLAY(sdata_var)
```

ELSE



Der ELSE-Befehl markiert das Ende des true-Teils des zuvor verknüpften if()-Befehls. Parsing-Befehle, die auf ELSE() folgen, werden ausgeführt, wenn FALSE das Ergebnis von IF() ist. Befehle werden bis zum nächsten entsprechenden ENDIF() ausgeführt.

Format

```
ELSE( )
```

Beispiel:

```
IF(i = 10)  
  ALERT("I is 10")  
ELSE()  
  ALERT("I is not 10")  
ENDIF()
```

Der direkte Vergleich mit einer negativen Zahl ist nicht möglich. Sie haben hierfür zwei Möglichkeiten:

- COMPARE der Parsing-Funktion verwenden
- Indirekten Vergleich wie folgt durchführen:

```
SET(i_compare_val=-10)  
IF(ivar > i_compare_val)  
  ALERT("ivar is greater than -10")  
endif()
```

ENCODE



Mit dem ENCODE-Befehl wird die Paketidentifizierung beibehalten. Dieser Befehl gleicht Byte (oder Zeichen) in Daten ab und versieht diese abgeglichenen Byte (Matched Bytes) mit einer Escape-Zeichenkette (durch Escape- oder Präfix-Vorgang). Die Escape-Zeichenkette wird in all den Fällen den abgeglichenen Byte vorangestellt, in denen diese Zeichen in den Daten gefunden werden.

Format

```
ENCODE(data_encode, match, escape)
```

Datentypen

Argument	Typ	Beschreibung
data_encode	svar (EINGABE/AUSGABE)	Die zu verschlüsselnde Zeichenkettendatenvariable. Das verschlüsselte Ergebnis wird wieder in diese Variable geschrieben.
match	Zeichenkette (EINGABE)	Die Zeichenkette von Byte zur Abstimmung in der data_encode-Zeichenkettenvariablen.
escape	Zeichenkette (EINGABE)	Die Escape-Zeichenkette, die in der data_encode-Variable jedem abgeglichenen Byte vorangestellt werden soll.

Beispiel:

Im nachfolgenden Beispiel werden zwei Datenzeichenketten so verschlüsselt, dass allen Leerschritten mit „#“ ein Präfix und allen ts und hs mit „!!“ ein anders Präfix vorangestellt wird.

```
COPY(data:"Preface all spaces with '#'")
ENCODE(data, " ", "#")
COPY(svar:"Preface 't's and 'h's with '!!'")
ENCODE(svar, "th", "!!")
```

Ergebnis:

```
data = "Preface# all# spaces# with# '#'"
svar = "Preface '!!t's and !!h's wi!!t!!h '!!'"
```

ENCODEMIME



Mit dem ENCODEMIME-Befehl kann der Benutzer eine Zeichenkette oder Zeichenkettenvariable mit der Base 64-Verschlüsselung verschlüsseln und die resultierende verschlüsselte Zeichenkette in einer Zeichenkettenvariablen speichern.

Format

```
ENCODEMIME(data, encoded_data)
```

Datentypen

Argument	Typ	Beschreibung
data	Zeichenkette/Zeichenkettenvariable (EINGABE)	Zu verschlüsselnde Datenzeichenkette.
encoded_data	Zeichenkettenvariable (AUSGABE)	Resultierende verschlüsselte Daten.

Beispiel:

```
COPY(s_data:"test encode64 command")
ENCODEMIME(s_data, s_endc_data)
```

Im obigen Beispiel wird mit dem ENCODEMIME-Befehl die Zeichenkette in der s_data-Variablen mithilfe der Base 64-Verschlüsselung verschlüsselt und die resultierende verschlüsselte Zeichenkette wird in s_endc_data gespeichert. S_endc_data wird folgendermaßen gefüllt:

```
VGvzdGluZyBEYXRhIEVudY29kaW5n
```

Ziehen Sie auch den [DECODEMIME](#)-Befehl zurate.

ENDFOR



Der ENDFOR-Befehl markiert das Ende des vorangegangenen for()-Blocks.

Format

```
ENDFOR( )
Beispiel
FOR(i=0,i<3,i=i+1)
ALERT("Still in loop")
ENDFOR( )
```

ENDIF



Der ENDIF-Befehl markiert das Ende des vorangegangenen if()-Blocks.

Format

```
ENDIF( )
```

Beispiel:

```
IF(i = 10)
ALERT("I is 10")
ELSE( )
ALERT("I is not 10")
ENDIF( )
```

Der direkte Vergleich mit einer negativen Zahl ist nicht möglich. Sie haben hierfür folgende Möglichkeiten:

- COMPARE der Parsing-Funktion verwenden
- Indirekten Vergleich wie folgt durchführen:

```
SET(i_compare_val=-10)  
IF(ivar > i_compare_val)  
  ALERT("ivar is greater than -10")  
ENDIF()
```

ENDWHILE



Der ENDWHILE-Befehl markiert das Ende des vorangegangenen while()-Blocks.

Format

```
ENDWHILE()  
  
Beispiel  
  
WHILE(i<3)  
  SET(i=i+1)  
ENDWHILE()
```

EVENT



Der EVENT-Befehl erstellt und sendet eine Warnmeldung. Er akzeptiert keine Parameter. Mit dem EVENT-Befehl wird die Warnmeldung automatisch anhand des Inhalts der reservierten Variablen erstellt.

Der Großteil der reservierten Variablen sind direkt den META-Tags der v3.2 Wizard-Schablone zugeordnet. Es werden nur die Variablen gesendet, die im Skript verwendet und nicht auf „eingestellt sind. Variablen wie „i_Severity“ und „s_Res“ sind zur Verarbeitung der Warnmeldung durch Collector Manager erforderlich.

Reservierte Ereignis-Variablen

HINWEIS: Wenn einer Kennung ein „e.“ vorangestellt ist, beispielsweise „e.crt“, dient dies als Verweis auf aktuelle Ereignisse. Wenn einer Kennung ein „w.“ vorangestellt ist, beispielsweise „w.crt“, dient dies als Verweis auf Verlaufsereignisse.

Variable	Kurzbeschreibung	Zuordnung zu META-Tag (Kennung)
s_BM	Base Message	Message (msg)
i_Severity	Severity	Severity (sev)
s_Res	Resource	Resource (res)
s_SubRes	SubResource	SubResource (sres)
s_ET	Event Time	EventTime (et)
s_P	Protocol	Protocol (prot)
s_DP	Destination Port	DestinationPort (dp)
s_SP	Source Port	SourcePort (sp)

Variable	Kurzbeschreibung	Zuordnung zu META-Tag (Kennung)
s_EVT	Event Name	EventName (evt)
s_SN	Sensor Name	SensorName (sn)
s_SIP	Source IP	Source IP (sip)
s_DIP	Destination IP	DestinationIP (dip)
s_SHN	Source Host Name	SourceHostName (shn)
s_DHN	Destination Host Name	DestinationHostName (dhn)
s_SUN	Source User Name	SourceUserName (sun)
s_DUN	Destination User Name	DestinationUserName (dun)
s_FN	File Name	FileName (fn)
s_EI	Extended Information	ExtendedInformation (ei)
s_RN	Reporter Name	ReporterName (rn)
s_ST	Sensor Type	Sensor Type (st)
s_PN	Product Name	ProductName (pn)
s_CRIT	Criticality	Criticality (crt)
s_VULN	Vulnerability	Vulnerability (vul)
s_CT1	Reserved Customer 1	Ct1 (ct1)
s_CT2	Reserved Customer 2	Ct2 (ct2)
s_CT3	Reserved Customer 3	Ct3 (ct3)
s_RT1	Device Attack Name (Reserved Sentinel 1)	Rt1 (rt1)
s_RT2	Reserved Sentinel 2	Rt2 (rt2)
s_RT3	Reserved Sentinel 3	Rt3 (rt3)
s_CV1 to s_CV100	Customer Variable 1 to 100 HINWEIS: 1 bis 10: long-Typ (Zahl) 11 bis 20: date-Typ 21 bis 100: string-Typ	Cv1 to Cv100 (cv1 to cv100)
s_RV1 to s_RV29	Reserved Value 1 to 29 HINWEIS: Für Novell reserviert.	Rv1 to Rv31 (rv1 to rv29)
s_RV30	AttackId	Rv30
s_RV31	DeviceName	Rv31
s_RV32	DeviceCategory	Rv32 (rv32)
s_RV33	EventContext	Rv33 (rv33)
s_RV34	SourceThreatLevel	Rv34 (rv34)
s_RV35	SourceUserContext	Rv35 (rv35)
s_RV36	DataContext	Rv36 (rv36)
s_RV37	SourceFunction	Rv37 (rv37)
s_RV38	SourceOperationalContext	Rv38 (rv38)
s_RV39	MSSPCustomerName	Rv39 (rv39)

Variable	Kurzbeschreibung	Zuordnung zu META-Tag (Kennung)
s_RV40 to s_RV43	Reserved Value 40 to 43 HINWEIS: Für Novell reserviert.	Rv40 to Rv43 (rv40 to rv43)
s_RV44	DestinationThreatLevel	Rv44 (rv44)
s_RV45	DestinationUserContext	Rv45 (rv45)
s_RV46	VirusStatus	Rv46 (rv46)
s_RV47	DestinationFunction	Rv47 (rv47)
s_RV48	DestinationOperationalContext	Rv48 (rv48)
s_RV49	ReservedVar49 HINWEIS: Für Novell reserviert.	Rv49 (rv49)
s_RV50	eSecTaxonomyLevel1	Rv50 (rv50)
s_RV51	eSecTaxonomyLevel2	Rv51 (rv51)
s_RV52	eSecTaxonomyLevel3	Rv52 (rv52)
s_RV53	eSecTaxonomyLevel4	Rv53 (rv53)
s_RV54 to s_RV100	Reserved Value 54 to 100 HINWEIS: Für Novell reserviert.	Rv54 to Rv100 (rv54 to rv100)

Automatische Formatierung

Für die reservierten Variablen „_DP“, „_SP“ und „_P“ wird die Kleinschreibung festgelegt, bevor die Ereignismeldung gesendet wird. Für die reservierten Variablen „_ST“ und „_PN“ wird die Großschreibung festgelegt, bevor die Ereignismeldung gesendet wird. „_ET“ der Ereigniszeitvariable wird festgelegt, wenn für hierfür im Standardzeitformat wie folgt keine Festlegung erfolgt:

s_Year-s_Month-s_Day~sHour:s_Min:s_Sec~s_AMPM24~s_TZ

Sie können diese Funktion außer Kraft setzen, indem Sie die s_ET-Variable mit weiteren Informationen festlegen. Minimal müssen sowohl „s_Hour“ als auch „s_Month“ festgelegt werden, damit die ET erstellt werden kann. Sämtliche leeren Felder werden im ET-Feld als NULL angegeben.

Reservierte Datum-/Uhrzeit-Variablen

Die ET-META-Tag-Variable „s_ET“ wird automatisch gefüllt, wenn für „s_ET“ keine Festlegung erfolgt und „s_Hour“ und „s_Month“ nicht leer sind. Die reservierten Datum-/Uhrzeit-Variablen sollten mit Werten festgelegt werden. Leere Felder werden als NULL angegeben. Für das s_Day-Feld werden die aus zwei Ziffern bestehenden Werte 01–09 formatiert. Der Skriptersteller kann den Monatswert mithilfe des [TRANSLATE](#)-Befehls und der Datei months.csv in eine zweistellige Zahl umwandeln. Hier die reservierten Datum-/Uhrzeit-Tags:

- s_Year
- s_Month
- s_Day
- s_Min
- s_Sec
- s_TZ

- s_Hour
- s_AMPM24

Reservierte Variablen für die Ereignissteuerung

Mithilfe zweier Variablen, „s_SendEITag“ und „s_SendETTag“ wird bestimmt, ob der EVENT-Befehl das EI- bzw. ET-Feld in eine Warnmeldung aufnimmt. Um das Senden dieser Felder zu deaktivieren, muss die Variable auf OFF eingestellt werden.

Format

```
EVENT ( )
```

Beispiel:

```
COPY(s_Res: "Resource")
SET(i_Severity = 3)
COPY(s_BM: "Alert")
EVENT( )
```

FILEA



Mit dem FILEA-Befehl wird der Inhalt einer Zeichenkette am Ende eines Flatfile auf dem Datenträger angefügt. Bei der Verwendung dieses Befehls gilt Folgendes:

- Der Dateiname wird mithilfe einer Zeichenkette angegeben
- Unter Windows verweist der Dateiname gemäß der Angabe auf die Datei, wenn der Dateiname mit einem Laufwerksbuchstaben, einem Doppelpunkt und einem umgekehrten Schrägstrich beginnt (z. B. c:\)
- Der vollständige Pfad der Datei sollte angegeben werden
- Wenn die Datei nicht vorhanden ist, wird sie erstellt
- Wenn die Datei nicht erstellt werden kann, hat der FILEA-Befehl keine Wirkung
- Die Datei wird geschlossen, nachdem ihr Daten angefügt wurden

Wenn Sie diesen Befehl als Teil eines Skripts schreiben, das von einem Collector ausgeführt werden soll, achten Sie unbedingt darauf, die richtige Pfadsyntax, einschließlich der Schrägstriche (/), zu verwenden. Denken Sie bei der Angabe des Pfads an den Escape-Vorgang für umgekehrten Schrägstrich und Schrägstrich. Die abschließende Null am Ende der Zeichenkette wird nicht in die Datei geschrieben.

Format

```
FILEA("filename", data)
```

Datentypen

Argument	Typ	Beschreibung
filename	Zeichenkette (EINGABE)	Der Name der Datei, auf die die Daten angewendet werden sollen.
data	Zeichenkette (EINGABE)	Die an die Datei anzufügende Datenzeichenkette.

Beispiel:

Im nachfolgenden Beispiel wird die Datei \temp\mux_data erstellt und der Inhalt von „s_variable“ wird der Datei hinzugefügt:

```
FILEA("c:\temp\mux_data", s_variable)
FILEA("mux_data", "literal")
FILEA("mux_data", s_variable)
```

Im nachfolgenden Beispiel wird am Ende der Revisionsprotokolldatei eine Zeichenkette hinzugefügt:

```
COPY(audit_str: "Sent 20 severity 5 alerts.")
FILEA("h:\temp\audit.log", audit_str)
```

FILEL



Mit dem FILEL-Befehl wird die Länge (in Byte) eines Flatfile abgerufen und der Wert wird in eine numerische Variable geschrieben. Bei der Verwendung dieses Befehls gilt Folgendes:

- Der Dateiname wird mithilfe einer Zeichenkette angegeben
- Unter Windows verweist der Dateiname gemäß der Angabe auf die Datei, wenn der Dateiname mit einem Laufwerksbuchstaben, einem Doppelpunkt und einem umgekehrten Schrägstrich beginnt (z. B. c:\)
- Wenn die Datei nicht vorhanden ist, hat der FILEL-Befehl keine Wirkung und der Inhalt von „numvar“ bleibt unverändert
- Die Datei wird geschlossen, nachdem die Daten aus ihr gelesen wurden

Wenn Sie diesen Befehl als Teil eines Skripts schreiben, das von einem Collector ausgeführt werden soll, achten Sie unbedingt darauf, die richtige Pfadsyntax, einschließlich der Schrägstriche (/), zu verwenden. Denken Sie bei der Angabe des Pfads an den Escape-Vorgang für umgekehrten Schrägstrich und Schrägstrich.

Format

```
FILEL("filename", i_length)
```

Datentypen

Argument	Typ	Beschreibung
filename	Zeichenkette (EINGABE)	Der Name der Datei, deren Länge bestimmt werden soll.
i_length	numvar (AUSGABE)	Die Länge der Datei (in Byte).

Beispiel:

```
FILEL("h:\tmp\onfotron.log", i_length)
```

Gibt die Länge der Datei infotron.log (in Byte) zurück. Beispiel:

```
i_length = 2390
```

FILER



Mit dem FILER-Befehl wird der Inhalt eines Flatfile auf dem Datenträger in eine Zeichenkettenvariable kopiert. Bei der Verwendung dieses Befehls gilt Folgendes:

- Der Dateiname wird mithilfe einer Zeichenkette angegeben
- Unter Windows verweist der Dateiname gemäß der Angabe auf die Datei, wenn der Dateiname mit einem Laufwerksbuchstaben, einem Doppelpunkt und einem umgekehrten Schrägstrich beginnt (z. B. c:\)
- Wenn die Datei nicht vorhanden ist, hat der FILER-Befehl keine Wirkung und der Inhalt von „svar“ bleibt unverändert
- Die Datei wird geschlossen, nachdem die Daten aus ihr gelesen wurden
- Optional können die maximal zu lesenden Byte eingegeben werden. Der max_bytes-Parameter kann nur in Kombination mit dem i_offset-Parameter verwendet werden.

Wenn Sie diesen Befehl als Teil eines Skripts schreiben, das von einem Collector ausgeführt werden soll, achten Sie unbedingt darauf, die richtige Pfadsyntax, einschließlich der Schrägstriche (/), zu verwenden. Denken Sie bei der Angabe des Pfads an den Escape-Vorgang für umgekehrten Schrägstrich und Schrägstrich.

Format

```
FILER("filename", dest, [i_offset [, i_max_bytes]])
```

HINWEIS: Der max_bytes-Parameter kann nur in Kombination mit dem i_offset-Parameter verwendet werden.

Datentypen

Argument	Typ	Beschreibung
filename	Zeichenkette (EINGABE)	Die Name der Datei, aus der die Datenzeichenkette gelesen werden soll.
data	svar (AUSGABE)	Die aus der Datei gelesenen Daten werden in diese Zeichenkettenvariable geschrieben.
i_offset	integer (EINGABE) [OPTIONAL]	Gibt eine Offset-Anzahl von Zeichen für den Beginn des Lesevorgangs an.
max_bytes	integer (EINGABE) [OPTIONAL]	Optional können die maximal zu lesenden Byte angegeben werden. <hr/> HINWEIS: Bei der Verwendung dieses Arguments muss das i_offset-Argument angegeben werden. <hr/>

Beispiel:

```
CLEAR(data)
FILEW("filename", data, 0, 20)
if(data = "")
ALERT(s_res_var, "Data file doesn't exist or is empty.",
      0)
ENDIF( )
```

FILEW



Mit dem FILEW-Befehl wird der Inhalt einer Zeichenkette in ein Flatfile auf dem Datenträger geschrieben. Bei der Verwendung dieses Befehls gilt Folgendes:

- Der bisherige Inhalt der Datei wird überschrieben
- Der Dateiname wird mithilfe einer Zeichenkette angegeben
- Unter Windows verweist der Dateiname gemäß der Angabe auf die Datei, wenn der Dateiname mit einem Laufwerksbuchstaben, einem Doppelpunkt und einem umgekehrten Schrägstrich beginnt (z. B. c:\)
- Wenn die Datei nicht vorhanden ist, wird sie erstellt
- Wenn die Datei nicht erstellt werden kann, hat der FILEW-Befehl keine Wirkung
- Die Datei wird geschlossen, nachdem die Daten in sie geschrieben wurden

Wenn Sie diesen Befehl als Teil eines Skripts schreiben, das von einem Collector ausgeführt werden soll, achten Sie unbedingt darauf, die richtige Pfadsyntax, einschließlich der Schrägstriche (/), zu verwenden. Denken Sie bei der Angabe des Pfads an den Escape-Vorgang für umgekehrten Schrägstrich und Schrägstrich.

Format

```
FILEW("filename", data)
```

Datentypen

Argument	Typ	Beschreibung
filename	Zeichenkette (EINGABE)	Die Name der Datei, in die die Datenzeichenkette geschrieben werden soll.
data	svar (AUSGABE)	Die Daten, die in die Datei geschrieben werden sollen.

Beispiel:

```
FILEW("filename", data)
FILEW("h:/\tmp/\infotron.stat", "SUCCESSFUL EXEC")
```

FOR



Mit dem FOR-Befehl kann der Steuerungsfluss als Schleife gestaltet werden. Bei der Verwendung dieses Befehls gilt Folgendes:

- Die Initialisierungsanweisung wird stets ausgeführt
- Wenn das Ergebnis der FOR()-Vergleichsanweisung „true“ ist, werden die Parsing-Befehle nach dem FOR() bis zum nächsten ENDFOR() ausgeführt. Die Inkrementierungsanweisung wird dann ausgeführt und der Steuerungsfluss gibt die Vergleichsanweisung zurück
- Wenn das Ergebnis des FOR()-Vergleichs „false“ ist, werden keine Parsing-Befehle zwischen FOR() und ENDFOR() ausgeführt. Die Inkrementierungsanweisung wird nicht ausgeführt
- Obwohl sämtliche Datentypen auf beiden Seiten der for()-Vergleichsanweisung zulässig sind, können numerische Werte nur mit numerischen Werten und Zeichenketten nur mit Zeichenketten verglichen werden
- Mögliche Operatoren für den FOR()-Vergleich: <, =, >, <=, >=, <>, &, + oder ^

Der direkte Vergleich mit einer negativen Zahl ist nicht möglich. Sie haben hierfür folgende Möglichkeiten:

- COMPARE der Parsing-Funktion verwenden
- Indirekten Vergleich wie folgt durchführen:

```
SET(i_compare_val=-10)  
FOR(ivar=0, ivar>i_compare_val, ivar=ivar-1)  
  ALERT("Still in loop")  
ENDFOR()
```

Format

```
FOR(initialization, compare, increment)
```

Datentypen

Argument	Typ	Beschreibung
initialization	SET() Parameter	Jeder zulässige Parameter, der an den SET()-Befehl übergeben werden kann. Ziehen Sie die SET()-Befehlsdefinition zurate.
conditional	IF() conditional	Jeder zulässige Parameter, der an den IF()-Befehl übergeben werden kann. Ziehen Sie die IF()-Befehlsdefinition zurate.
increment	SET() Parameter	Jeder zulässige Parameter, der an den SET()-Befehl übergeben werden kann. Ziehen Sie die SET()-Befehlsdefinition zurate.

Beispiel:

```
FOR(i=0, i<3, i=i+1)
```

GETCONFIG



Ruft die aktuelle Einstellung für eine Systemeigenschaft ab. Mit diesem Befehl werden Systemeigenschaften abgerufen, die mithilfe des [SETCONFIG](#)-Befehls festgelegt wurden. Diese Befehle werden zum Festlegen von Variablen und zum Abrufen aktueller Werte für Systemeigenschaften verwendet, die sich u. U. regelmäßig ändern, etwa eine Protokolldatei, die jeden Tag in das aktuelle Datum umbenannt wird.

Folgende Systemeigenschaften sind verfügbar:

Systemeigenschaften	Beispiel
▪ System.OS.Family	Solaris und Windows
▪ System.OS.Name	Windows 2000
▪ System.OS.Version.Major	5
▪ System.OS.Version.Minor	0
▪ System.Net.Hostname	ESECServer
▪ System.Net.IP_List	Durch Strichpunkt getrennte Liste der IPs für diesen Host. Beispiel: "172.163.3.45;172.45.2.1"

Ziehen Sie auch den [SETCONFIG](#)-Befehl zurate.

Zwei Parameter sind erforderlich.

- Mit dem ersten erforderlichen Parameter wird die Konfigurationsoption (FileConnector.InputFile) oder (FileConnector.OutputFile) definiert.
- Mit dem zweiten erforderlichen Parameter wird der abzurufende Konfigurationswert definiert.

Format

```
GETCONFIG(Config Option, Value)
```

Datentypen

Argument	Typ	Beschreibung
Config Option	Zeichenkette (EINGABE)	Name der abzurufenden Konfigurationsvariablen. Eingabedatei = "FileConnector.InputFile" Ausgabedatei = "FileConnector.OutputFile"
Value	Zeichenkette (EINGABE)	Abzurufende Konfigurationseinstellung.

Beispiel:

```
GETCONFIG("FileConnector.InputFile", s_inputfilename)  
GETCONFIG("FileConnector.OutputFile", s_outputfilename)
```

Aktueller Inhalt von Ausgabevariablen

```
"C:\filename.txt"
```

GETENV



Mit dem GETENV-Befehl wird der Wert einer Umgebungsvariablen abgerufen.

Format

```
GETENV(Environment Key, Variable to store value)
```

Datentyp

Argument	Typ	Beschreibung
Environment Key	Zeichenkette (EINGABE)	Name der Umgebungsvariablen.
Variable to store value	Zeichenkettenvariable (EINGABE)	Ziel, an dem die Umgebungsvariable gespeichert wird.

Beispiel:

```
GETENV("ESEC_HOME", s_EsecHome)
```

HEXTONUM



Mit dem HEXTONUM-Befehl wird eine Hex-Zeichenkette mit bis zu 4 Byte Hex-Daten in eine Dezimalzahl umgewandelt und die Dezimalzahl wird in eine Ganzzahl- oder Float-Variable geschrieben. Mehr als 4 Byte führen zu ungültigen Daten.

Format

```
HEXTONUM(bytes_data, i_val [, [-]i_4] [, ioffset])
```

Datentypen

Argument	Typ	Beschreibung
bytes_data	Zeichenkette (EINGABE)	Zeichenkette mit 1 bis 4 Byte. (Beispiel: „\FF“, „\FF FF“, „\3C 4A F2“, „\43 76 F3 FF“ oder „TEST“). Die Hex-Zahl, die durch diese Byte dargestellt wird, wird in einen Ganzzahlwert, i_val, umgewandelt.
i_val	numvar (AUSGABE)	Das Dezimal-Äquivalent der Hex-Zahl wird in diese Variable, ivar oder fvar, geschrieben.

Argument	Typ	Beschreibung
i_len	Numerisch (EINGABE) [OPTIONAL]	Anzahl der Hex-Byte, die in eine Ganzzahl umgewandelt werden sollen (der Absolutwertbereich muss zwischen 1 und 4 liegen). Wenn Sie diesen Parameter nicht festlegen, ist der Standardwert die Anzahl der Byte in der Eingabezeichenkette, bytes_data (maximal 4 Byte). Wenn „i_len“ positiv ist, werden die Byte von links nach rechts interpretiert (Most Significant Byte nach Least Significant Byte). Wenn „i_num_bytes“ negativ ist, werden die Byte von rechts nach links interpretiert (Least Significant Byte nach Most Significant Byte).
ioffset	Numerisch (EINGABE) [OPTIONAL]	Offset-Anzahl der in „bytes_data“ zu überspringenden Byte.

Beispiel:

Im nachfolgenden Beispiel werden die Daten in der Hex-Zeichenkette "\5A32\" in einen Ganzzahlwert umgewandelt, der von MSB nach LSB und dann von LSB nach MSB interpretiert wird.

```
COPY(data: "\5A 32\")
HEXTONUM(data, ivar1)
HEXTONUM(data, ivar2, -2)
```

HINWEIS: Bei der Hex-Ersetzung wird mit \0000\ eine Zeichenkette beendet; folglich wird aus „xxxx\0000\yyyy“ „xxxx“.

Aktueller Inhalt von Ausgabevariablen:

```
ivar1 = 23090
ivar2 = 12890
```

IF



Mit dem IF-Befehl werden zwei Werte miteinander verglichen.

- Wenn das Ergebnis der IF()-Anweisung „true“ ist, werden die Parsing-Befehle nach dem IF() bis zum nächsten ELSE() oder ENDIF() ausgeführt.
- Wenn das Ergebnis der IF()-Anweisung „false“ ist, werden die Parsing-Befehle nach dem ELSE() bis zum nächsten ENDIF() ausgeführt.
- Wenn kein ELSE() verwendet wird, werden zwischen dem IF() und ENDIF() keine Parsing-Befehle ausgeführt (wenn das Ergebnis der IF()-Anweisung „false“ ist).
- Obwohl sämtliche Datentypen auf beiden Seiten der IF()-Anweisung zulässig sind, können numerische Werte nur mit numerischen Werten und Zeichenketten nur mit Zeichenketten verglichen werden

- Mögliche Operatoren für den IF()-Vergleich: <, =, >, <=, >=, <>, &, + oder ^.
Verwenden Sie den logischen NOT-Operator (^) nicht gemeinsam mit einer Zeichenkettenvariablen. Anderenfalls tritt ein Syntaxfehler auf.

Der direkte Vergleich mit einer negativen Zahl ist nicht möglich. Sie haben hierfür folgende Möglichkeiten:

- COMPARE der Parsing-Funktion verwenden
- Indirekten Vergleich wie folgt durchführen:
SET(i_compare_val=-10)
IF(ivar > i_compare_val)
ALERT("ivar is greater than -10")
ENDIF()

Format

IF(<expr>)

Hierbei gilt:

```
expr ::= var
      | (<expr>)
      | ^ <expr>
```

Hierbei muss <expr> zu einer Ganzzahl- oder Float-Variablen werden.

```
| <expr> <|=|>|<=|>=|<>|&|+ <expr>
```

Hierbei müssen die beiden <expr> zum selben Typ werden.

Datentypen

Argument	Typ	Beschreibung
data1	Variable (EINGABE)	Die Daten, die mit „data2“ verglichen werden sollen. Wenn „data2“ nicht verwendet wird, wird daraus ein logischer Operator (0 = false, Rest = true).
logical operator	< = > <= >= <> & + ^	Kleiner Gleich Größer Kleiner oder gleich Größer oder gleich Nicht gleich Logischer AND-Operator Logischer OR-Operator Logischer NOT-Operator
data2	Beliebig (EINGABE) [OPTIONAL]	Die Daten, die mit „data1“ verglichen werden sollen. Müssen denselben Typ wie „data1“ aufweisen.
...	Wie oben	Mit bis zu 200 einzelnen Parametern können komplexe logische Ausdrücke erstellt werden.

Beispiel:

```
IF(s = "test" & i_count < 5)
script(test)
ELSE()
IF((i <= i_num) + (i_count <> 10) & (i_page))page("111")
ENDIF()
ENDIF()
```

INC



Mit dem INC-Befehl wird eine numerische Variable um 1 erhöht. Bei der Verwendung dieses Befehls muss entweder eine Ganzzahl- oder eine Float-Variable angegeben werden.

Format

```
INC(i_counter)
```

Datentypen

Argument	Typ	Beschreibung
i_counter	numvar (EINGABE/AUSGABE)	Die numerische Variable, die um 1 erhöht werden soll.

Beispiel:

```
SET(icounter = 0)
INC(icounter)
INC(icounter)
```

Ergebnis:

```
icounter = 2
```

INDICATOR



Mit dem INDICATOR-Befehl werden Indikatormeldungen an Sentinel gesendet. Diese Meldungen enthalten Text, der in Sentinel im angegebenen Indikator angezeigt werden soll.

Format

```
INDICATOR(name, value)
```

HINWEIS: Vor Version 4.0 enthielt der INDICATOR-Befehl zusätzliche Argumente, die nicht mehr verwendet werden. Um die Kompatibilität mit älteren Collectors zu gewährleisten, wird bei diesen Argumenten im Wizard-Befehlseditor „Not Used“ angegeben.

Datentypen

Argument	Typ	Beschreibung
name	Zeichenkette (EINGABE)	Name des Indikators.
value	Zeichenkette (EINGABE)	In Sentinel Console anzuzeigender Indikatortext. Beispiel: PRINTER ON

Beispiel:

```
INDICATOR("memory", "5 MB")  
INDICATOR(name, value)
```

HINWEIS: Der Indikatorname im Parsing-Befehl muss mit dem Indikatornamen in Sentinel übereinstimmen, da andernfalls der Indikator in Sentinel Console nicht aktualisiert wird.

INFO_CLEAR_TAGS



Mit dieser Funktion werden sämtliche Variablen auf null eingestellt (Zeichenketten werden hingegen gelöscht), die Teil des Infoblock-Satzes sind, auf den die Zugriffsnummer verweist. Mit [INFO_CONSTANT_TAGS](#) kann dies für eine Teilmenge dieser Tags verhindert werden.

Format

```
INFO_CLEAR_TAGS(<IN handle>)
```

Datentypen

Argument	Typ	Beschreibung
IN handle	Zeichenkette (EINGABE)	Typ des Informationsblocks

INFO_CLOSE



Mit diesem Befehl wird eine Infoblock-Sitzung geschlossen. Wenn er aufgerufen wird, werden zunächst sämtliche nicht gesendeten Infoblocks gesendet, genau wie dies mit dem INFO_SEND-Befehl der Fall wäre. Anschließend wird eine Meldung zum Schließen der Infoblock-Sitzung gesendet, indem das EOD-(End Of Data-)Attribut des infos-Elements auf „true“ eingestellt wird. Nachdem die Meldung zum Schließen gesendet wurde, wird die Segmentnummer („segnum“) um 1 erhöht.

Format

```
INFO_CLOSE(<IN handle>)
```

Datentypen

Argument	Typ	Beschreibung
IN handle	Zeichenkette (EINGABE)	Typ des Informationsblocks

INFO_CONSTANTTAGS



Mit diesem Befehl können Tags angegeben werden, die beim Aufruf von [INFO_CLEAR_TAGS](#) nicht gelöscht werden sollen. Durch die Übergabe von null oder mehr Tagnamen wird der Satz konstanter Tags erstellt. Beim mehrmaligen Aufruf dieser Funktion wird die Liste der konstanten Tags zurückgesetzt.

Format

```
INFO_CONSTANTTAGS(<IN handle>, [<IN tag name>, ...])
```

Datentypen

Argument	Typ	Beschreibung
IN handle	Zeichenkette (EINGABE)	Typ des Informationsblocks
IN tag name	Zeichenkette (EINGABE)	Name für den Verweis auf „IN handle“

INFO_CREATE



Hiermit wird ein neuer Informationsblock-Satz erstellt. Sie müssen eine Zugriffsnummer übergeben (die Sie in jedem weiteren Befehl verwenden, um diesen Informationsblock-Satz zu ändern). Sie müssen außerdem einen Typ übergeben. Hierbei handelt es sich um eine Zeichenkette Ihrer Wahl, sie sollten jedoch bestimmte Formatvorgaben erfüllen (siehe [INFO_SEND](#)).

Wenn Sie [INFO_CREATE](#) für eine bereits vorhandene Zugriffsnummer aufrufen, wird der Inhalt bei dieser Zugriffsnummer so gelöscht, als hätten Sie eine neue Zugriffsnummer erstellt. Sie müssen [INFO_SETTAG](#) und [INFO_CONSTANTTAGS](#) erneut aufrufen.

Format

```
INFO_CREATE(<OUT handle>,<IN type>)
```

Datentypen

Argument	Typ	Beschreibung
OUT handle	Zeichenkette (AUSGABE)	Name für den Verweis auf „IN type“
IN type	Zeichenkette (EINGABE)	Typ des Informationsblocks

INFO_DUMP



Dieser Befehl sorgt dafür, dass der aktuelle Zustand des Infoblock-Satzes in einer Zeichenkettenvariablen permanent in eine Zeichenkettenvariable geschrieben wird. Dieser Befehl wurde zu Testzwecken aufgenommen, kann jedoch auch zur Wiedergabe von Informationsblock-Sätzen bzw. zu deren Speicherung in einer Textdatei oder einem anderen Dateityp Ihrer Wahl verwendet werden. Der Unterschied zu [INFO_SEND](#) besteht zudem darin, dass der aktuelle Status nicht gelöscht wird.

Format

```
INFO_DUMP(<IN handle>, <OUT string-variable>)
```

Datentypen

Argument	Typ	Beschreibung
IN handle	Zeichenkette (EINGABE)	Typ des Informationsblocks
OUT string-variable	Zeichenkette (AUSGABE)	Zeichenkettenvariable für den Verweis auf „IN handle“

INFO_PUSH



Hiermit wird der Tagvorgang für die aktuellen Werte aller Tagnamen (über die verknüpften Variablen) durchgeführt und sie werden an das Ende der Liste mit Infoblock-Sätzen verschoben, auf die durch eine Zugriffsnummer verwiesen wird. Blöcke werden so lange im Satz gesammelt, bis sein Inhalt durch den Aufruf von [INFO_CREATE](#), [INFO_SEND](#) oder [INFO_CLOSE](#) gelöscht wird. Für INFO_CREATE erfolgt keine Aktion. Für INFO_SEND werden die Infoblöcke an „Collector Manager“ gesendet. Für INFO_CLOSE werden die Infoblöcke an „Collector Manager“ gesendet; zudem wird eine Meldung zum Schließen des Infoblocks (EndOfData oder EOD) gesendet.

Format

```
INFO_PUSH(<IN handle>)
```

Datentypen

Argument	Typ	Beschreibung
IN handle	Zeichenkette (EINGABE)	Typ des Informationsblocks

INFO_SEND



Hiermit wird der aktuelle Satz mit Infoblöcken über einen Kommunikationskanal gesendet; dieser Kanal wird durch den Typ angegeben, der bei [INFO_CREATE](#) verwendet wurde (wird an das Wort „infoblock.“ angefügt, einschließlich Punkt). Beispiel: Wenn „vulnerability“ der Typ ist, trägt der Kanal, über den die Meldung gesendet wird, den Namen „infoblock.vulnerability“.

Zudem wird mit diesem Befehl der aktuelle Satz an Infoblöcken gelöscht und die Segmentnummer („segnum“) wird um 1 erhöht.

Format

```
INFO_SEND(<IN handle>)
```

Datentypen

Argument	Typ	Beschreibung
IN handle	Zeichenkette (EINGABE)	Typ des Informationsblocks

INFO_SETTAG



Mit diesem Befehl erfolgt die Bindung einer Skriptvariablen an den Namen eines Attributs. Wenn INFO_PUSH aufgerufen wird (siehe [INFO_PUSH](#)), werden sämtliche an diesen Befehl gebundenen Variablen als Attribute in einem Blockeintrag festgelegt.

Format

```
INFO_SETTAG(<IN handle, IN tag name, IN variable>)
```

Datentypen

Argument	Typ	Beschreibung
IN handle	Zeichenkette (EINGABE)	Typ des Informationsblocks
IN tag name	Zeichenkette (EINGABE)	Typ des Tagnamens
IN variable	Zeichenkette (EINGABE)	Typ der Variable

Tags für Anfälligkeits-Infoblöcke

Nachfolgend sind zulässige Tags für Anfälligkeits-Infoblöcke für den INFO_SETTAG-Befehl aufgeführt. Die als erforderlich gekennzeichneten Tags müssen festgelegt werden, damit die Infoblöcke als Anfälligkeit (vulnerability) gespeichert werden. Selbst wenn der Infoblock nicht als Anfälligkeit gespeichert wird, werden die als konstant markierten Tags dennoch aus dem Infoblock extrahiert. Wenn ein Tag festgelegt wird, das in der nachfolgenden Liste nicht enthalten ist, wird es vom Anfälligkeits-Backend ignoriert.

Name des Tags	Erklärung	Typ	Konstant	Erforderlich
ScannerInstance	Der Name, den der Benutzer für das Absuchprogramm (Scanner) vergibt. Die Angabe erfolgt üblicherweise in den Collector-Parametern.	Zeichenkette	X	
ProductName	Name des Absuchprogramms.	Zeichenkette	X	
ProductVersion	Version des Absuchprogramms.	Zeichenkette	X	
ScannerType	Der Typ des Absuchprogramms.	Zeichenkette	X	
Vendor	Der Name des Herstellers des Absuchprogramms.	Zeichenkette	X	
ScanType	TEILWEISE oder VOLLSTÄNDIG	Zeichenkette	X	
ScanStartDate	Der Beginn des Absuchvorgangs	Zeichenkette		
ScanEndDate	Das Ende des Absuchvorgangs	Zeichenkette		
IP	IP der Ressource	Zeichenkette		X
HostName	Der Hostname der Ressource	Zeichenkette		
Location	Der Ort, an der sich die Ressource befindet	Zeichenkette		
Department	Die Abteilung, in der sich die Ressource befindet	Zeichenkette		
BusinessSystem	Das Geschäftssystem der Ressource	Zeichenkette		
OperationalEnvironment	Die Betriebsumgebung der Ressource	Zeichenkette		
Regulation	Die Bestimmung der Ressource	Zeichenkette		
RegulationRating	Das Bestimmungs-Rating der Ressource	Zeichenkette		
Criticality	Die Gefährlichkeit der Ressource [1–25]	Zahl		
VulnModule	Das Modul zur Erkennung der Anfälligkeit	Zeichenkette		
PortNumber	Die Portnummer der Anfälligkeit	Zahl		

Name des Tags	Erklärung	Typ	Konstant	Erforderlich
PortName	Der Name des Ports der Anfälligkeit	Zeichenkette		
NetworkProtocol	Das Netzwerkprotokoll der Anfälligkeit	Zahl		
ApplicationProtocol	Das Anwendungsprotokoll der Anfälligkeit	Zeichenkette		
AssignedVulnSeverity	Der zugewiesene Anfälligkeitsschweregrad.	Zahl		
ComputedVulnSeverity	Der berechnete Anfälligkeitsschweregrad.	Zahl		
VulnDescription	Die Beschreibung der Anfälligkeit.	Zeichenkette		
VulnSolution	Die Lösung für die Anfälligkeit.	Zeichenkette		
VulnSummary	Die Lösung für die Anfälligkeit.	Zeichenkette		
VulnCrossRefs	Eine Liste mit Codes für die Anfälligkeit.	Zeichenkette		
DetectedOs	Das Betriebssystem, das erkannt wurde, als die Anfälligkeit bemerkt wurde	Zeichenkette		
DetectedOsVersion	Die Betriebssystemversion, die erkannt wurde, als die Anfälligkeit bemerkt wurde.	Zeichenkette		
ScannedApp	Die Anwendung, die erkannt wurde, als die Anfälligkeit bemerkt wurde	Zeichenkette		
ScannedAppVersion	Die Anwendungsversion, die erkannt wurde, als die Anfälligkeit bemerkt wurde	Zeichenkette		
VulnUserName	Der Benutzername für die Anfälligkeit.	Zeichenkette		
VulnUserDomain	Die Domäne des Benutzers der Anfälligkeit.	Zeichenkette		
VulnTaxonomy	Die Taxonomie der Anfälligkeit.	Zeichenkette		
ScannerClassification	Die vom Absuchprogramm angegebene Anfälligkeitsklassifizierung.	Zeichenkette		
ExtendedInformation	Erweiterte Informationen, die gemeinsam mit dieser Anfälligkeit gespeichert werden können	Zeichenkette		
VulnName	Der vom Absuchprogramm vergebene Name für die Anfälligkeit.	Zeichenkette		

INFO_*-Befehlsbeispiel

Sentinel teilt Anfälligkeitsabsuchvorgänge in kleinere Bestandteile (Infoblock-Sitzungen) auf, die leichter verarbeitet werden können. Eine Infoblock-Sitzung besteht aus mehreren Infoblock-Sätzen mit einer jeweils um 1 höheren Segmentnummer („segnum“), auf die eine Meldung zum Schließen der Infoblock-Sitzung folgt. Der Verweis auf eine Instanz einer Infoblock-Sitzung erfolgt durch die zugehörige, global eindeutige ID. Bei jedem Aufruf von INFO_SEND wird ein Infoblock mit den Werten aus dem aktuellen Push-Vorgang und der aktuellen Segmentnummer („segnum“) gesendet. Unmittelbar nach dem Senden des Infoblock-Satzes wird die segnum um 1 erhöht. INFO_SEND wird für jeden Datenstapel aufgerufen; anschließend wird der INFO_CLOSE-Befehl zum Schließen der Infoblock-Sitzung aufgerufen. Die Meldung zum Schließen des Infoblocks besteht aus einem Infoblock-Satz, bei dem das Attribut-EOD auf „true“ eingestellt ist.

Beispiel:

```
INFO_CREATE(h_vuln, "vulnerability")
INFO_SETTAG(h_vuln, "ALPHA", s_alpha)
INFO_SETTAG(h_vuln, "BETA", i_beta)
INFO_SETTAG(h_vuln, "GAMMA", s_gamma)
INFO_SETTAG(h_vuln, "DELTA", i_delta)
INFO_SETTAG(h_vuln, "^1E*P$S I(L)O.N--", f_epsilon)
INFO_CONSTANTTAGS(h_vuln, "GAMMA", "DELTA", "^1E*P$S
    I(L)O.N--")
SET(i_beta=12345)
SET(i_delta=123456789)
SET(f_epsilon=1.234)
COPY(s_alpha:"a is for apple")
COPY(s_gamma:"c is for coffee")
INFO_PUSH(h_vuln)
INFO_CLEAR_TAGS(h_vuln)
INFO_PUSH(h_vuln)
INFO_DUMP(h_vuln, s_simulate)
INFO_SEND(h_vuln)
SET(i_beta=6789)
SET(i_delta=987654321)
SET(f_epsilon=3.1415926)
COPY(s_alpha:"a is for acorn")
COPY(s_gamma:"c is for carrot")
INFO_PUSH(h_vuln)
INFO_SEND(h_vuln)
INFO_CLOSE(h_vuln)
```

Ergebnis:

```
<?xml version="1.0" encoding="UTF-8"?>
<infos id="B008961E00CB1026B8F000065BBD13AB"
  type="vulnerability" segnum="0" version="4.2.0.0"
  EOD="false">
  <info ALPHA="a is for apple" BETA="12345"
    DELTA="123456789" GAMMA="c is for coffee"
    _1EPSILON="1.234"/>
  <info ALPHA="" BETA="0" DELTA="123456789" GAMMA="c is for
    coffee" _1EPSILON="1.234"/>
</infos>
<?xml version="1.0" encoding="UTF-8"?>
<infos id="B008961E00CB1026B8F000065BBD13AB"
  type="vulnerability" segnum="1" version="4.2.0.0"
  EOD="false">
  <info ALPHA="a is for acorn" BETA="6789"
    DELTA="987654321" GAMMA="c is for carrot"
    _1EPSILON="3.1415926"/>
</infos>
<?xml version="1.0" encoding="UTF-8"?>
<infos id="B008961E00CB1026B8F000065BBD13AB"
  type="vulnerability" segnum="2" version="4.2.0.0"
  EOD="true">
</infos>
```

IPTONUM



Mit dem IPTONUM-Befehl wird eine Zeichenkettendarstellung einer IPv4-Adresse in eine Ganzzahl umgewandelt und die Ganzzahl wird in eine Ganzzahlvariable geschrieben. Von dieser Funktion werden lediglich IPv4-Adressen unterstützt. Eine IPv4-Adresse, die nicht innerhalb des zulässigen Bereichs liegt, führt zu ungültigen Daten.

Format

```
IPTONUM(ip_address, i_integer, i_valid)
```

Datentypen

Argument	Typ	Beschreibung
ip_address	svar (EINGABE)	IPv4-Zeichenkettenadresse.
i_integer	numerisch (AUSGABE)	IPv4-Zeichenkettenadresse wird in einen Ganzzahlwert umgewandelt. Der Ganzzahlwert wird in diese Variable geschrieben.
i_invalid	ivar (AUSGABE) [OPTIONAL]	0 bedeutet, dass die IP-Adresse ungültig ist. 1 bedeutet, dass die IP-Adresse gültig ist.

Beispiel:

Im nachfolgenden Beispiel wird die IPv4-Adresse "10.10.10.255" in eine Ganzzahl umgewandelt. „i_valid“ wird auf 1 eingestellt, was auf ein gültiges Ergebnis hinweist.

```
IPTONUM("10.10.10.255", i_y, i_valid)
```

Aktueller Inhalt der Ausgabevariablen:

```
i_y = 168430335  
i_valid = 1
```

Im nachfolgenden Beispiel wird die ungültige IPv4-Adresse "10.10.10.258" in die Ganzzahl 0 umgewandelt. „i_valid“ wird auf 0 eingestellt, was auf ein ungültiges Ergebnis hinweist.

```
IPTONUM("10.10.10.258", i_y, i_valid)
```

Aktueller Inhalt der Ausgabevariablen:

```
i_y = 0  
i_valid = 0
```

Mit dem NUMTOIP-Befehl wird eine Zahl in eine IP-Adresse umgewandelt. Weitere Informationen finden Sie unter [NUMTOIP](#).

LENGTH oder LENGTH-OPTION2



Mit dem LENGTH-Befehl wird eine numerische Variable anhand der Länge in Byte einer Zeichenkettenvariablen (ohne die abschließende Null) festgelegt.

HINWEIS: Im visuellen Editor von Collector Builder werden LENGTH und LENGTH-OPTION2 als separate Befehle aufgelistet. Es handelt sich hierbei um ein- und denselben Befehl. Sie werden als Beschreibungen von Varianten desselben Befehls zur Verfügung gestellt. Wenn Sie LENGTH-OPTION2 im Texteditor verwenden möchten, geben Sie LENGTH ein.

Format

```
LENGTH(i_length, s_variable)
```

Datentypen

Argument	Typ	Beschreibung
s_variable	Zeichenkette (EINGABE)	Die Zeichenkette (normalerweise eine Zeichenkettenvariable) in der die Länge berechnet wird.
i_length	numvar (AUSGABE)	Die Länge der Zeichenkettenvariablen, s_variable, wird in diese numerische Variable geschrieben.

Beispiel:

```
LENGTH(i_length, source)
LENGTH(i_num_bytes, "It makes no sense to do this, as we
    know the string whose length we are checking")
```

Ergebnis:

```
i_num_bytes = 80
```

LOOKUP



Mit dem LOOKUP-Befehl werden im Empfangspuffer bzw. in einer Zeichenkette gefundene Daten mit Schlüsselzeichenketten abgeglichen, die in einer angegebenen Such-Schlüsseldatei gefunden wurden.

Wenn ein Datensatz gefunden wird, der Byte für Byte mit den Daten übereinstimmt, werden die Parsing-Befehle im Such-Schlüsseldateidatensatz verarbeitet.

Wenn eine Zeichenkette als erster Parameter im LOOKUP-Befehl angegeben wird, verwendet der LOOKUP-Befehl diese Zeichenkette für die Suche nach der Such-Schlüsseldatei.

Dieser Befehl umfasst fünf Argumente bzw. Parameter.

- compare – Wenn ein numerischer Wert als dieser Parameter angegeben wird, wird diese Anzahl an Byte (der numerische Wert) Daten aus dem Empfangspuffer (beginnend bei der Rx Buffer Pointer Position) als Zeichenkette verwendet, wenn der Vergleich mit den Schlüsselzeichenketten der Such-Schlüsseldatei erfolgt.
- lookup name – Dieser Parameter gibt den Namen der Such-Schlüsseldatei relativ zum WORKBENCH_HOME-Verzeichnis an.
- imatch – Eine optionale Ganzzahlvariable, die angegeben werden kann und die den Status des LOOKUP-Befehls zurückgibt. (0 = keine Übereinstimmung gefunden, 1 = Übereinstimmung gefunden).
- parameter file – Ein optionaler Parameter, bei dem es sich um den Namen einer Parameterdatei handelt, die nicht die standardmäßige Parameterdatei ist. Der standardmäßige Parameterdateiname lautet <Collector>.par. Dieser Dateiname sollte das .par-Suffix nicht enthalten.
- column name – Eine optionaler Parameter, bei dem es sich um die Spalte der Parameterdatei handelt, die für Suchwerte verwendet werden soll. Der Standardspaltenname ist der Schablonenname. Bei der Angabe dieses Parameters muss auch ein Parameterdateiname angegeben werden.

Format

```
LOOKUP(compare, lookup filename [, imatch] [, [parameter  
filename] [, column name]])
```

Datentypen

Argument	Typ	Beschreibung
compare	Zeichenkette (EINGABE) oder Numerisch (EINGABE)	Die Daten, die für den Vergleich anhand der Felder in der Such-Schlüsseldatei verwendet werden sollen. Der Vergleich erfolgt in diesem Fall Byte für Byte. Die Anzahl der Byte aus dem Empfangspuffer (unter Verwendung der aktuellen Rx Buffer Pointer Position), die für den Vergleich anhand der Felder in der Such-Schlüsseldatei verwendet werden sollen. Der Vergleich erfolgt in diesem Fall Byte für Byte. <hr/> HINWEIS: Dieser Vorgang ist nur möglich, wenn der Empfangspuffer mit „rxbuff“ festgelegt wurde. <hr/>
lookup filename	Zeichenkette (EINGABE)	Der Name der Such-Schlüsseldatei
imatch	numvar (AUSGABE) [OPTIONAL]	Es wurde eine Übereinstimmung gefunden. 0 = Nein 1 = Ja
Parameter filename	Zeichenkette (EINGABE)	Der Parameterdateiname. Standard: Collector.par
column name	Zeichenkette (EINGABE)	Die zu verwendende Spalte in der Parameterdatei. Standard: Collector name

Beispiel:

```
LOOKUP(data, filename, imatch)
```

Im nachfolgenden Beispiel wird der Dateiname „key_01“ anhand des in die Parameterdatei, nicht in die Such-Schlüsseldatei, geschriebenen Namen ermittelt.

```
LOOKUP(s_variable, {key_01})  
LOOKUP(s_variable, {key_01}, imatch, "Send One Alert",  
"GeoElements")
```

Wenn Parameterdefinitionen in der Suchdatei enthalten sind, finden Sie sie in der Send One Alert-Parameterdatei in der GeoElements-Spalte.

NEGSEARCH



Mit dem NEGSEARCH-Befehl wird eine rückwärtsgerichtete Suche nach einer Zeichenkette im Empfangspuffer durchgeführt. Dieser Befehl umfasst zwei Parameter.

- **search** – Die Suche beginnt an der aktuellen Rx Buffer Pointer Position und wird rückwärtsgerichtet fortgesetzt, bis die Zeichenkette gefunden wird bzw. der Anfang des Empfangspuffer erreicht ist. Wenn die Zeichenkette im Rahmen des Suchvorgangs gefunden wird, wird der Rx Buffer Pointer dahingehend aktualisiert, dass er auf das erste Byte der Suchzeichenkette verweist. Wenn die Zeichenkette im Rahmen des Suchvorgangs nicht gefunden wird, bleibt der Rx Buffer Pointer unverändert.
- **ifound** – Ein optionaler Parameter; es handelt sich um eine Ganzzahlvariable, die auf 1 eingestellt wird, wenn die Zeichenkette im Rahmen des Suchvorgangs gefunden wird; wenn die Zeichenkette im Rahmen des Suchvorgangs nicht gefunden wird, wird sie auf 0 eingestellt.

Format

```
NEGSEARCH(search[, ifound])
```

Datentypen

Argument	Typ	Beschreibung
search	Zeichenkette (EINGABE)	Die gesuchte Zeichenkette im Empfangspuffer; die Suche beginnt an der aktuellen Rx Buffer Pointer Position und wird rückwärtsgerichtet durchgeführt.
ifound	numvar (AUSGABE) (OPTIONAL)	Gibt an, ob die gesuchte Zeichenkette gefunden wurde. 0 = Nicht gefunden 1 = Gefunden

Beispiel:

```
NEGSEARCH("MINOR ALARM")  
NEGSEARCH(search_string)
```

In den nachfolgenden Beispielen wird nach einer Zeilenschaltung und einem Zeilenvorschub gesucht:

```
NEGSEARCH("\0d0a\" )  
NEGSEARCH(data, ifound)
```

Weiteres Beispiel:

Der Buchstabe mit Unterstrich gibt die aktuelle Rx Buffer Pointer Position, also die Position des Empfangspufferzeigers, im Beispiel an.

HINWEIS: Bei der Hex-Ersetzung wird mit \0000\ eine Zeichenkette beendet; folglich wird aus „xxxx\0000\yyyy“ „xxxx“.

```
Rx Buffer = "Minor Alarm Radio A"  
NEGSEARCH("Ala")
```

Ergebnis:

```
Rx Buffer = "Minor Alarm Radio A"
```

NUMTOHEX



Mit dem NUMTOHEX-Befehl wird ein numerischer Wert in Hex-Daten umgewandelt und diese Hex-Byte (bis zu 4 Byte) werden dann in eine Zeichenkette geschrieben.

Format

```
NUMTOHEX(i_decimal, hex_data)
```

Datentypen

Argument	Typ	Beschreibung
i_decimal	Numerisch (EINGABE)	Ganzzahlwert, der in Hex-Daten umgewandelt werden soll.
hex_data	svar (AUSGABE)	Zeichenkette mit 1 bis 4 Byte, die als Hex-Byte vorliegen (die Angabe erfolgt durch den numerischen Wert, i_decimal).

Beispiel:

Im nachfolgenden Beispiel wird die Dezimalzahl 16777215 in Hex-Daten umgewandelt.

```
SET(i_decimal = 16777215)
NUMTOHEX(i_decimal, shex)
```

Aktueller Inhalt der Ausgabevariablen:

```
shex = "\ff ff ff\"
```

NUMTOIP



Mit dem NUMTOIP-Befehl wird ein numerischer Wert in eine IPv4-Adresse umgewandelt und diese IP-Adresse dann in eine Zeichenkette geschrieben.

Format

```
NUMTOIP(i_integer, ip_address)
```

Datentypen

Argument	Typ	Beschreibung
i_integer	numerisch (EINGABE)	Ganzzahlwert, der in IPv4-Adresse umgewandelt werden soll.
ip_address	svar (AUSGABE)	IPv4-Zeichenkettenadresse

Beispiel:

Im nachfolgenden Beispiel wird die Dezimalzahl 16777215 in eine IPv4-Adresse umgewandelt.

```
SET(i_integer = 167772161)
NUMTOIP(i_integer, s)
```

Aktueller Inhalt der Ausgabevariablen:

```
s = "10.0.0.1"
```

Mit dem IPTONUM-Befehl wird eine IP-Adresse in eine Zahl umgewandelt. Weitere Informationen finden Sie unter [IPTONUM](#).

PARSER_ATTACHVARIABLE



Mit dem PARSER_ATTACHVARIABLE-Befehl kann der Name eines Name-Wert-Paars mit einer target_variable verknüpft werden.

In den meisten Fällen empfiehlt es sich, einen Parser (ein Analyseprogramm) zu erstellen und eine Variable im Initialisierungszustand außerhalb der Schleife anzufügen. Im Anschluss kann dieser Parser durch die Verwendung in der Parsing-Schleife erneut verwendet werden.

Für verwandte Parsing-Befehle ziehen Sie den [PARSER_CREATEBASIC](#)-Befehl und den [PARSER_PARSESTRING](#)-Befehl zurate.

NVP-(Name-Value Pair-)Parser

Der nachfolgende Codeauszug dient der Veranschaulichung des NVP-Parser (Analyseprogramm für Name-Wert-Paare):

```
PARSER_CREATEBASIC (h_nvp, "nvp", "separator==",
    "entry_separator= ", "value_quotes=/\"",
    value_quotes_optional=yes")
PARSER_ATTACHVARIABLE (h_nvp, "this", s_this)
PARSER_ATTACHVARIABLE (h_nvp, "me", s_me)
PARSER_ATTACHVARIABLE (h_nvp, "hello", s_hello)
PARSER_PARSESTRING (h_nvp, "this=/\"that/\" me=/\"you =
    them/\" hello=/\"goodbye/\"")
```

Parameter

Die nachfolgend angegebenen Parameter werden erkannt, wenn sie in folgendem Format vorliegen:

```
"<parameter>=<value>"
```

<parameter> steht für eines der unten angegebenen Elemente und <value> für einen entsprechenden Wert für diesen Parameter.

- separator – Das Zeichen, mit dem der Name vom Wert abgetrennt wird
- entry_separator – Das Zeichen, mit dem ein Name-Wert-Paar vom nachfolgenden Paar abgetrennt wird
- name_quotes – Das Zeichen, in das der Name gestellt wird (z. B. „, oder ')

- `value_quotes` – Das Zeichen, in das der Wert gestellt wird
- `name_quoted` – Wird auf „yes“ eingestellt, damit der NVP-Parser die `name_quotes`-Option beachtet
- `value_quoted` – Wird auf „yes“ eingestellt, damit der NVP-Parser die `value_quotes`-Option beachtet
- `name_quotes_optional` – Wird auf „yes“ eingestellt, um optionale Anführungszeichen für den Namen zuzulassen. Wenn dieser Parameter auf „yes“ eingestellt ist und keine Anführungszeichen verwendet werden, wird der Name durch ein optionales Leerzeichen gefolgt vom Trennzeichen beendet.
- `value_quotes_optional` – Wird auf „yes“ eingestellt, um optionale Anführungszeichen für den Wert zuzulassen

Wenn dieser Parameter auf „yes“ eingestellt ist und keine Anführungszeichen verwendet werden, wird der Wert durch ein optionales Leerzeichen gefolgt von „`entry_separator`“ beendet.

Format

```
PARSER_ATTACHVARIABLE(<parser_handle>, <name>,  
                      <target_variable>)
```

Datentypen

Argument	Typ	Beschreibung
<code>parser_handle</code>	Zeichenkettenvariable (EINGABE)	Die Zugriffsnummervariable eines erstellten Parser.
<code>name</code>	Zeichenkette (EINGABE)	Der Name eines Name-Wert-Paars.
<code>target_variable</code>	Beliebige Variable (AUSGABE)	Die Variable, die mit dem Wert festgelegt wird, der mit den Namen eines Name-Wert-Paars verknüpft ist.

Hier ein Checkpoint-Parser-Beispiel:

```
COLLECTOR SETUP STATE:
PARSER_CREATEBASIC(h_nvp, "nvp", "separator==",
                  "entry_separator= ", "value_quotes=/",
                  "value_quotes_optional=yes")
PARSER_ATTACHVARIABLE(h_nvp, "action", s_EVT)
PARSER_ATTACHVARIABLE(h_nvp, "d_port", s_DP)
PARSER_ATTACHVARIABLE(h_nvp, "proto", s_P)
PARSER_ATTACHVARIABLE(h_nvp, "src", s_SIP)
PARSER_ATTACHVARIABLE(h_nvp, "dst", s_DIP)

PARSE STATE:
PARSER_PARSESTRING(h_nvp, s_RXBufferString)
```

PARSER_CREATEBASIC



Mit dem PARSER_CREATEBASIC-Befehl wird ein Parser definiert und mit einer parser_handle verknüpft. Weitere Informationen finden Sie unter [NVP \(Name-value Pair\) Parser](#) unter [PARSER_ATTACHVARIABLE](#).

In den meisten Fällen empfiehlt es sich, einen Parser (ein Analyseprogramm) zu erstellen und eine Variable im Initialisierungszustand außerhalb der Schleife anzufügen. Im Anschluss kann dieser Parser durch die Verwendung in der Parsing-Schleife erneut verwendet werden.

Für einen weiteren verwandten Parsing-Befehl ziehen Sie den [PARSER_PARSESTRING](#)-Befehl zurate.

Format

```
PARSER_CREATEBASIC(<parser_handle>, <parser_name>, [,  
    <nvp> [, ...]])
```

Datentypen

Argument	Typ	Beschreibung
parser_handle	Zeichenkettenvariable (AUSGABE)	Die Variable, mit der ab diesem Punkt auf diesen Parser verwiesen wird.
parser_name	Zeichenkette (EINGABE)	Der Zeichenkettenname des einfachen Parser, den Sie erstellen. HINWEIS: Gegenwärtig wird nur „nvp“ erkannt.
nvp	Zeichenkette (EINGABE) (OPTIONAL)	Das Name-Wert-Paar. Null oder mehr Zeichenketten, die einen Eigenschaftsnamen gefolgt von einem Gleichheitszeichen gefolgt von einem Wert enthalten. Welche Parameter erkannt werden, wird durch den ausgewählten parser_name bestimmt. HINWEIS: Wenn der Parser-Name auf „nvp“ eingestellt ist, müssen folgende Argumente verwendet werden: "separator==" "entry_separator=" " "value_quotes=/" "value_quotes_optional=yes"
nvp1	Zeichenkette (EINGABE) (OPTIONAL)	Name-Wert-Paar 1.
nvp2	Zeichenkette (EINGABE) (OPTIONAL)	Name-Wert-Paar 2.

Argument	Typ	Beschreibung
...	Zeichenkette (EINGABE) (OPTIONAL)	Weitere Name-Wert-Paare.

Ein Beispiel finden Sie unter [Checkpoint Parser example](#) unter [PARSER_ATTACHVARIABLE](#) (Datentyp).

PARSER_NEXT



Mit dem PARSER_NEXT-Befehl wird der Parser an die nächste Stelle in der Analysezeichenkette bewegt; hierbei werden die vom Befehl [PARSER_ATTACHVARIABLE](#) festgelegten Variablen ausgefüllt.

Format

```
PARSER_NEXT(<parser_handle>, <success_flag>)
```

Datentyp

Argument	Typ	Beschreibung
parser_handle	Zeichenkette Variable (EINGABE)	Die Zugriffsnummervariable eines erstellten Parser.
success_flag	numvar (EINGABE)	0: Nicht erfolgreicher Analysevorgang 1: Erfolgreicher Analysevorgang

PARSER_PARSESTRING



Mit dem PARSER_PARSESTRING-Befehl wird die string_to_parse mit dem erstellten Parser verarbeitet, auf den die parser_handle verweist. Auf diese Weise kann jede beliebige Zeichenkette für den Analysevorgang (Parsing) erstellt werden, ohne dass hierfür eine Datenstromquelle bzw. der Empfangspuffer zwingend erforderlich sind.

Für weitere Informationen ziehen Sie den [PARSER_ATTACHVARIABLE](#)-Befehl und den [PARSER_CREATEBASIC](#)-Befehl zurate.

Die reservierte s_RXBufferString-Variable kann als string_to_parse verwendet werden, und zwar nach dem Empfangsstatus zur Analyse der Skripteingabe. Weitere Informationen finden Sie unter [NVP \(Name-value Pair\) Parser](#) unter [PARSER_ATTACHVARIABLE](#).

Format

```
PARSER_PARSESTRING(<parser_handle>, <string_to_parse>)
```

Datentypen

Argument	Typ	Beschreibung
parser_handle	Zeichenkette Variable (EINGABE)	Die Zugriffsnummervariable eines erstellten Parser.
string_to_parse	Zeichenkette (EINGABE)	Die einzelne Zeichenkette, die diesen Parser durchläuft.

Ein Beispiel finden Sie unter [Checkpoint Parser example](#) unter [PARSER_ATTACHVARIABLE](#) (Datentyp).

PAUSE



Mit dem PAUSE-Befehl wird das aktuelle Skript sofort für „n“ Sekunden angehalten. Der PAUSE-Befehl kann zwischen Anweisungen in einer Parsing-Phase sowie zwischen Phasen verwendet werden. Der PAUSE-Befehl ist besonders beim Festlegen der Polling-Zyklusdauer bzw. dann hilfreich, wenn sichergestellt werden soll, dass das Polling nicht zu schnell erfolgt (z. B. beim Polling eines Datenbankprotokolls).

Während des Analysierens können mehrere PAUSE-Befehle festgelegt werden.

Format

```
PAUSE(iseconds)
```

Argument	Typ	Beschreibung
iseconds	Numerisch (EINGABE)	Anzahl der Pause in Sekunden vor dem Übergang zur nächsten Phase.

Beispiel:

```
PAUSE(10)
PAUSE(iseconds)
```

Oder

```
IF(slowing=true)
    pause(50)
ENDIF( )
```

POPUP



Mit dem POPUP-Befehl wird der Inhalt einer Zeichenkette in einem Textfenster angezeigt, in dem Sie blättern können.

Format

```
POPUP(data [, title])
```

Datentypen

Argument	Typ	Beschreibung
data	Zeichenkette (EINGABE)	Die Datenzeichenkettenmeldung, die im Popup-Fenster angezeigt werden soll.
title	Zeichenkette (EINGABE) [OPTIONAL]	Die Zeichenkette, die als Titel des Popup-Fensters verwendet werden soll (Standard = „Popup DATA“).

Beispiel:

```
POPUP(data)
POPUP("Hello World", "Title String")
POPUP(data, title)
```

PRINTF



Mit dem PRINTF-Befehl werden formatierte Daten in eine Zeichenkettenvariable (svar) kopiert. Der PRINTF-Befehl ist ein erweiterter Parsing-Befehl. Wenn Sie mit der Parsing-Befehlssprache noch nicht vertraut sind, empfiehlt sich u. U. die Verwendung der Befehle [COPY](#) und [APPEND](#), bis Sie über die entsprechenden Fertigkeiten verfügen.

Bei der Verwendung dieses Befehls gilt Folgendes:

- Es muss eine svar als Zielzeichenkette angegeben werden.
- Es muss eine Formatzeichenkette angegeben werden.
- Durch die Angabe optionaler zusätzlicher Parameter kann das Absuchen basierend auf der Formatzeichenkette erfolgen.

Formatzeichenkette

Halten Sie sich an folgende Konvention, um HEX-Daten in der Formatzeichenkette verwenden zu können:

```
\HX HX HX\
```

Wenn am Ende der Formatzeichenkette ein Zeilenvorschub eingefügt werden soll, muss die Formatzeichenkette wie folgt aussehen:

```
Format String\0a\
```

Die Formatzeichenkette für eine Zeilenschaltung ist \0d0a\; Beispiel:

```
PRINTF(message, "Voltage is %lf \0d0a\ ", f_volts)
```

Die Formatzeichenkette für ein Tabulatorzeichen \09\; Beispiel:

```
PRINTF(message, "Voltage = \09\ %lf", f_volts)
```

Format

```
PRINTF(dest, format [, <paramList>])
```

Hierbei gilt:

```
<paramList> ::= var [, <paramList>]
```

Datentypen

Argument	Typ	Beschreibung
dest	svar (AUSGABE)	Die Zielzeichenkettenvariable, in die die formatierte Zeichenkette geschrieben werden soll.
format	Zeichenkette (EINGABE)	Das Format der Zeichenkette, die in die Zielzeichenkettenvariable kopiert werden soll. Ähnelt dem Format des C printf-Befehls; Beispiel: "Looping %d in %s" (siehe „% Characters for Output Format“).
parm1	Beliebig (EINGABE) [OPTIONAL]	Sämtliche Datentypen mit Ausnahme von Array. Muss mit der Formatierungszeichenkette übereinstimmen.
parm2	Beliebig (EINGABE) [OPTIONAL]	Sämtliche Datentypen mit Ausnahme von Array. Muss mit der Formatierungszeichenkette übereinstimmen.
...	Beliebig (EINGABE) [OPTIONAL]	Sämtliche Datentypen mit Ausnahme von Array. Muss mit der Formatierungszeichenkette übereinstimmen.

Format

% Characters for Output Format

Zeichen	Typ	Ausgabeformat
%d	Ganzzahl	Dezimalganzzahl mit Vorzeichen.
%le	Float-Zahl	Wert mit Vorzeichen im Format [-]d.ddd e [sign]ddd ...hierbei steht „d“ für eine einzelne Dezimalstelle, „ddd“ für eine oder mehrere Dezimalstellen, „ddd“ für genau drei Dezimalstellen und „sign“ ist+ oder -.
%lf	Float-Zahl	Wert mit Vorzeichen im Format [-]dddd.dddd ...hierbei steht „ddd“ für eine oder mehrere Dezimalstellen. Die Anzahl der Stellen vor dem Dezimalzeichen hängt von der Größe der Zahl ab, die Anzahl der Stellen nach dem Dezimalzeichen von der gewünschten Genauigkeit.
%lg	Float-Zahl	Wert mit Vorzeichen, der im f- oder -Format ausgegeben wird, je nachdem, welches Format für den jeweiligen Wert und die gewünschte Genauigkeit am kompaktesten ist. Das e-Format wird nur verwendet, wenn der Exponent des Werts kleiner als -4 bzw. größer oder gleich dem precision-Arguments ist. Nachfolgende Nullen werden abgeschnitten und das Dezimalzeichen wird nur angezeigt, wenn darauf mindestens eine Ziffer folgt.
%s	Zeichenkette	Ausgabe einer Zeichenkettenvariablen.

Anzeigen von Genauigkeitsstellen

Standardmäßige wird mit dem PRINTF-Befehl eine Fließkommazahl mit einer Genauigkeit von sechs Stellen angezeigt. Der Standard für die Genauigkeit von sechs Stellen trifft auch auf Doppelgenauigkeitszahlen zu.

Wenn zusätzliche Stellen angezeigt werden sollen, geben Sie in der PRINTF()-Formatspezifikation einen Wert im Feld für die Genauigkeit an:

```
%[<width>][.<precision>] type>
```

Beispiel:

```
PRINTF(dest, "%2.3lf", fvar)
```

Die Ausgabe sieht in diesem Fall wie folgt aus: 22.012, also zwei Stellen links vom Dezimalzeichen und 3 Stellen rechts vom Dezimalzeichen.

In den nachfolgenden Beispielen wird die Übergabe von Zeichenketten- und Ganzzahlvariablen erläutert.

```
PRINTF(dest,format_string) PRINTF(mystring,
    "val of matrix[%d][%d] = %s",
    index_x, index_y, matrix[index_x][index_y])
PRINTF(dest,"Looping %d in state %s",iloop,state)
PRINTF(dest,"Formatted %s Data into
    %s","string","dest")
```

Im nachfolgenden Beispiel wird die Übergabe einer Float-Variable an eine Zeichenkette erläutert.

```
PRINTF(message,"Voltage is %lf",f_volts)
```

Verwenden Sie zur Ausgabe von Fließkommazahlen „%lf“ oder „%le“.

REGEXPREPLACE



Mit dem REGEXPREPLACE-Befehl werden Zeichenketten unter Verwendung regulärer Ausdrücke gesucht und ersetzt. Wenn die Zeichenkette im Rahmen des Suchvorgangs gefunden wird, wird die regexpreplace-Zeichenkette ersetzt. Mit dem REGEXPREPLACE-Befehl wird ein globaler Ersetzungsvorgang vorgenommen, es wird also nicht nur das erste Vorkommen ersetzt.

Format

```
REGEXPREPLACE(dest_string, search, replace)
```

Datentypen

Argument	Typ	Beschreibung
dest_string	svar (EINGABE/AUSGABE)	Die Zeichenkettenvariable, in der Byte ersetzt werden.

Argument	Typ	Beschreibung
search	Zeichenkette (EINGABE) oder svar (EINGABE/AUSGABE)	Die zu ersetzende Suchzeichenkette.
replace	Zeichenkette (EINGABE) Oder svar (EINGABE/AUSGABE)	Die Austauschzeichenkette; kann eine Länge von null aufweisen, um auf eine Null-Zeichenkette hinzuweisen.

Beispiel:

```
COPY(string:"The 1st time")
REGEXREPLACE(string, "1st", "2nd")
```

Ergebnis:

```
string = "The 2nd time"
```

HINWEIS: In diesem Beispiel kann ein regulärer Ausdruck die "1st"-Zeichenkette ersetzen.

Ersetzung durch Null-Zeichenkette

```
COPY(string:"The 1st time")
REGEXPREPLACE(string, "1st", "")
```

Ergebnis:

```
string="The time"
```

Weitere Informationen zu regulären Ausdrücken und dem Portable Character Set (PCS) finden Sie unter „Reguläre Ausdrücke“.

Sentinel verwendet eine mit POSIX (Portable Operating System Interface for UNIX) kompatible Bibliothek für reguläre Ausdrücke. POSIX setzt sich aus IEEE-(Institute of Electrical & Electronics Engineers-) und ISO-(Industry Standards Organisation-)Standards zusammen, mit denen die Kompatibilität zwischen POSIX-fähigen Betriebssystemen gewährleistet wird; hierzu zählt der Großteil der UNIX-Varianten.

REGEXPSEARCH, REGEXPSEARCH_EXPLICIT bzw. REGEXPSEARCH_STRING



Mit dem REGEXPSEARCH-Befehl wird im Empfangspuffer (Rx Buffer) bzw. in der angegebenen Eingabezeichenkettenvariablen mithilfe von regulären Ausdrücken in einem vorwärtsgerichteten Suchvorgang nach einer Zeichenkette gesucht. Ausdrucksgruppen werden ebenfalls unterstützt.

HINWEIS: Im visuellen Editor von Collector Builder werden REGEXPSEARCH, REGEXPSEARCH_EXPLICIT bzw. REGEXPSEARCH_STRING als separate Befehle aufgelistet. Es handelt sich hierbei um ein- und denselben Befehl. Sie werden als Beschreibungen von Varianten desselben Befehls zur Verfügung gestellt. Wenn Sie REGEXPSEARCH_EXPLICIT oder REGEXPSEARCH_STRING im Texteditor verwenden möchten, geben Sie REGEXPSEARCH ein.

Empfangspuffer

Der Suchvorgang innerhalb des Empfangspuffers verläuft wie folgt:

- Die Suche beginnt an der aktuellen Rx Buffer Pointer Position und wird vorwärtsgerichtet fortgesetzt, bis die Zeichenkette gefunden wird bzw. das Ende des Empfangspuffer erreicht ist.
- Wenn die Zeichenkette im Rahmen des Suchvorgangs gefunden wird, wird der Rx Buffer Pointer dahingehend aktualisiert, dass er auf das erste Byte der Zeichenkette verweist, nach der gesucht wurde. Die Rx Buffer Pointer Position wird auch beim Übergang in einen anderen Zustand/eine andere Phase beibehalten, es sei denn, es erfolgt die explizite Änderung mithilfe des RESET-Befehls.
- Wenn die Zeichenkette im Rahmen des Suchvorgangs nicht gefunden wird, bewegt sich der Rx Buffer Pointer nicht.

Wenn dieser Befehl zum Durchsuchen des Empfangspuffers verwendet wird, handelt es sich beim optionalen zweiten Parameter um eine Ganzzahlvariable, die auf 1 eingestellt wird, wenn die Zeichenkette im Rahmen des Suchvorgangs gefunden wird; wenn die Zeichenkette im Rahmen des Suchvorgangs nicht gefunden wird, wird sie auf 0 eingestellt.

Zeichenkettenvariable

Der Parse-(Analyse-)Zeiger wird von Zeichenkettenvariablen nicht unterstützt, folglich ergibt sich für die Suche in einer Zeichenkettenvariablen eine andere Dynamik. Das Muster regulärer Ausdrücke stimmt entweder teilweise oder ganz mit der Eingabezeichenkette überein. Wenn die Konfiguration des Musters regulärer Ausdrücke mithilfe von Ausdrucksgruppen erfolgt, kann der Eingabezeichenketteninhalt, der mit den Ausdrucksgruppen übereinstimmt, in Ausgabevariablen gespeichert werden. Es gibt zwei Ausdrucksgruppen-Ausgabeoptionen. Die eine besteht darin, die Liste mit Variablen gemäß der Reihenfolge der Ausdrucksgruppen zu füllen, die andere besteht darin, ein Zeichenkettenarray anzugeben.

Wenn der reguläre Ausdruck mit der Eingabezeichenkettenvariablen übereinstimmt, wird eine angegebene Liste mit Variablen bzw. ein angegebenes Ausgabearray mit den Gruppenwerten festgelegt und die gefundene Variable wird auf einen um 1 höheren Wert als die Anzahl der Gruppen eingestellt; wenn keine Übereinstimmung vorliegt, wird sie auf 0 eingestellt.

Wenn es sich bei der Ausgabe der Gruppenwerte um ein Zeichenkettenarray handeln soll, enthält das erste mit „0“ indizierte Element die Übereinstimmungszeichenkette. Die Übereinstimmungszeichenkette enthält den Inhalt, der mit dem gesamten regulären Ausdruck übereingestimmt hat, unabhängig von Ausdrucksgruppen. Der Inhalt der ersten Ausdrucksgruppe wird folglich an einer mit „1“ indizierten Arrayposition gespeichert. Beachten Sie beim Durchlaufen des Ausgabearrays als Schleife, dass der `i_Found_Tokens`-Wert das erste Element (die Übereinstimmungszeichenkette) ausgleicht, indem er jeweils um 1 höher als die Gesamtgruppenzahl liegt. In einer `for`-Schleife kann die Stoppbedingung (geringer als `i_Found_Tokens`-Wert) weiterhin verwendet werden, höchstwahrscheinlich beginnen Sie mit dem Index jedoch bei „1“, nicht bei „0“.

Wenn Sie angeben, dass die Gruppenwerte in einer Liste mit Ausgabevariablen gespeichert werden sollen, nicht in einem Array, kann mit diesem Befehl die Typumwandlung erfolgen. Obwohl es sich bei der Eingabezeichenkette um eine Zeichenkette handelt, kann es sich bei Komponenten mit der Zeichenkette um Ziffern handeln. Wenn diese Ziffern als Ganzzahlen oder Fließkommawerte behandelt werden sollen, erfolgt die Umwandlung, wenn für die Ausgabevariablen der richtige Typ angegeben wird.

Einfacher REGEX-Abgleich

Ausdruck	Beschreibung
.	Beliebiges Zeichen
\d	Beliebige Ziffer
\w	Beliebiges alphanumerisches Zeichen
\s	Beliebiges Leerzeichen
+	1 oder mehr des Vorangegangenen
*	0 oder mehr des Vorangegangenen

Format

Als Empfangspuffer:

```
REGEXPSEARCH(search[, ifound])
```

Als Zeichenkettenvariable:

```
REGEXPSEARCH(Input_String, s_Regular_Exp_Pattern,
    i_Found_Tokens[, s_Output_Results[]])
REGEXPSEARCH(s_Input_String, s_Regular_Exp_Pattern,
    i_Found_Tokens, s_Match[, var1, var2, ...])
```

Datentypen

Argument	Typ	Beschreibung
s_Input_String	Zeichenkette oder Zeichenkettenvariable (EINGABE) [OPTIONAL]	Die Zeichenkette oder Zeichenkettenvariable, die nach in regex angegebenen regex-Übereinstimmungen durchsucht werden soll.
s_Regular_Exp_Pattern	Zeichenkette (EINGABE)	Die Zeichenkette, nach der im Empfangspuffer (ab der aktuellen Rx Buffer Pointer Position vorwärts) gesucht werden soll; weitere Möglichkeiten sind Eingabezeichenkettenliteral und Eingabezeichenkettenvariable.
i_Found_Tokens	numvar (AUSGABE) [OPTIONAL]	Gibt an, ob die gesuchte Zeichenkette gefunden wurde. 0: Muster regulären Ausdrucks stimmt nicht überein 1: Muster regulären Ausdrucks stimmt überein, jedoch mit keiner der angegebenen Ausdrucksgruppen 2: Muster regulären Ausdrucks stimmt mit 1 angegebenen Ausdrucksgruppe überein N+1: Muster regulären Ausdrucks stimmt mit n angegebenen Ausdrucksgruppen überein <hr/> HINWEIS: Die Variable „I_found_tokens“ kann als Test für die Übereinstimmung verwendet werden, da der Wert nicht null ist, wenn der reguläre Ausdruck übereinstimmt.
s_Match	Zeichenkette (AUSGABE) [BEDINGT]	Wird nur bei Musterübereinstimmung gefüllt und muss angegeben werden, wenn eine Liste mit Ausdrucksgruppen-Ausgabevariablen verwendet wird. Wenn die Gruppenwerte in einem Ausgabearray gespeichert werden, ist „s_Match“ KEIN zulässiger Parameter.
Variablenliste ODER s_Output_Results[]	Alle sind zulässig (AUSGABE) [OPTIONAL] ODER Zeichenkettenarray (AUSGABE) [OPTIONAL]	Die Liste der Variablen, in die die Gruppenwerte geschrieben werden sollen. Der Wert entspricht der Zuordnung in der Reihenfolge, in der Gruppenwerte angegeben wurden (Beachtung der Vorrangsregeln).

In den nachfolgenden Beispielen wird im Empfangspuffer nach einer Zeilenschaltung und einem Zeilenvorschub gesucht:

```
REGEXPSEARCH( "\0d0a\" )
```

Im nachfolgenden Beispiel wird im Empfangspuffer nach dem Wort „alarm“ gesucht:

```
REGEXPSEARCH( "alarm" )
```

HINWEIS: Bei der Hex-Ersetzung wird mit \0000\ eine Zeichenkette beendet; folglich wird aus „xxxx\0000yyyy“ „xxxx“.

Detailliertes Beispiel der Suche nach einem Muster in einem Literal-Zeichenkettenwert:

```
REGEXPSEARCH( "2003 Jan 15 13:34:20",  
    "(/\\d+)/\\s+(/\\w+)/\\s+(/\\d+)/\\s+(/\\d+):( /\\d+):( /\\d+)",  
    i_Success, s_Match, s_Year, s_Month, s_Day, s_Hour,  
    s_Minute, s_Second)
```

Hierbei gilt:

```
i_Success = 7  
s_Match = 2003 Jan 15 13:34:20  
s_Year = 2003  
s_Month = Jan  
s_Day = 15  
s_Hour = 13  
s_Minute = 34  
s_Second = 20
```

Weitere Informationen zu regulären Ausdrücken und dem Portable Character Set (PCS) finden Sie in Kapitel 2 im Abschnitt „Reguläre Ausdrücke“.

Sentinel verwendet eine mit POSIX (Portable Operating System Interface for UNIX) kompatible Bibliothek für reguläre Ausdrücke. POSIX setzt sich aus IEEE-(Institute of Electrical & Electronics Engineers-) und ISO-(Industry Standards Organisation-)Standards zusammen, mit denen die Kompatibilität zwischen POSIX-fähigen Betriebssystemen gewährleistet wird; hierzu zählt der Großteil der UNIX-Varianten.

REPLACE



Mit dem REPLACE-Befehl werden Zeichenketten gesucht und ersetzt.

Wenn die Zeichenkette im Rahmen des Suchvorgangs gefunden wird, wird die replace-Zeichenkette ersetzt. Mit dem REPLACE-Befehl wird ein globaler Ersetzungsvorgang vorgenommen, es wird also nicht nur das erste Vorkommen ersetzt.

Format

```
REPLACE(dest_string, search, replace)
```

Datentypen

Argument	Typ	Beschreibung
dest_string	svar (EINGABE/AUSGABE)	Die Zeichenkettenvariable, in der Byte ersetzt werden.
search	Zeichenkette (EINGABE)	Die zu ersetzende Suchzeichenkette.
replace	Zeichenkette (EINGABE)	Die Austauschzeichenkette.

Beispiel:

```
COPY(string:"The 1st time")  
REPLACE(string, "1st", "2nd")
```

Ergebnis:

```
string = "The 2nd time"
```

HINWEIS: In diesem Beispiel kann ein regulärer Ausdruck die „1st“-Zeichenkette ersetzen.

RESET



Mit dem RESET-Befehl wird der Rx Buffer Pointer auf 0 zurückgesetzt.

Format

```
RESET( )
```

Das ^-Symbol gibt beispielsweise die Rx Buffer Pointer Position, also die Position des Empfangspufferzeigers, an.

```
rxbuff = "abcdefg"  
          ^  
  
RESET( )
```

Ergebnis:

```
"abcdefg"  
  ^
```

RXBUFF



Mit dem RXBUFF-Befehl wird der Empfangspuffer mit dem Inhalt einer Zeichenkette bzw. Zeichenkettenvariablen mit Anführungszeichen überschrieben. Der Inhalt des Empfangspuffers ändert sich umgehend und Rx Buffer Pointer und verwalteter Wert werden auf 0 zurückgesetzt.

Format

```
RXBUFF( s_data )
```

Datentypen

Argument	Typ	Beschreibung
s_data	Zeichenkette (EINGABE)	Die Datenzeichenkette, die in den Empfangspuffer geschrieben werden soll. Bei dieser Zeichenkette handelt es sich umgehend um die neue Empfangspufferzeichenkette.

Beispiel:

Im nachfolgenden Beispiel wird mit dem [FILER](#)-Befehl eine Datei namens alert.data gelesen und der Inhalt dieser Datei in eine Zeichenkettenvariable namens s_data geschrieben. Bei diesem Beispiel gilt folgende Annahme:

```
alert.data: "Minor Alarm Xterminal A"
```

Dann werden mit dem RXBUFF-Befehl diese Daten in den Empfangspuffer geschrieben, genau so, als ob die Daten über einen Port eingegangen wären.

```
FILER("alert.data", s_data)
RXBUFF(s_data)
//copies data from Rx BUFFER into S_Alarm_Priority,
    stopping before the string "Alarm")
COPY(S_Alarm_Priority:," Alarm")
```

Ergebnis:

```
S_Alarm_Priority= "Minor"
```

SEARCH



Mit dem SEARCH-Befehl wird im Empfangspuffer (Rx Buffer) ein vorwärtsgerichteter Suchvorgang nach einer Zeichenkette durchgeführt.

Die Suche verläuft wie folgt:

- Die Suche beginnt an der aktuellen Rx Buffer Pointer Position und wird vorwärtsgerichtet fortgesetzt, bis die Zeichenkette gefunden wird bzw. das Ende des Empfangspuffer erreicht ist.
- Wenn die Zeichenkette im Rahmen des Suchvorgangs gefunden wird, wird der Rx Buffer Pointer dahingehend aktualisiert, dass er auf das erste Byte der Zeichenkette verweist, nach der gesucht wurde. Die Rx Buffer Pointer Position wird auch beim Übergang in einen anderen Zustand/eine andere Phase beibehalten, es sei denn, es erfolgt die explizite Änderung mithilfe des RESET-Befehls.
- Wenn die Zeichenkette im Rahmen des Suchvorgangs nicht gefunden wird, bewegt sich der Rx Buffer Pointer nicht.

Bei der Verwendung dieses Befehls handelt es sich beim optionalen zweiten Parameter um eine Ganzzahlvariable, die auf 1 eingestellt wird, wenn die Zeichenkette im Rahmen des Suchvorgangs gefunden wird; wenn die Zeichenkette im Rahmen des Suchvorgangs nicht gefunden wird, wird sie auf 0 eingestellt.

Format

```
SEARCH(search[, ifound])
```

Datentypen

Argument	Typ	Beschreibung
search	Zeichenkette (EINGABE)	Die Zeichenkette, nach der im Empfangspuffer gesucht werden soll (ab der aktuellen Rx Buffer Pointer Position vorwärts).
ifound	numvar (AUSGABE) [OPTIONAL]	Gibt an, ob die gesuchte Zeichenkette gefunden wurde. 0 = Nicht gefunden 1 = Gefunden

Beispiel:

In den nachfolgenden Beispielen wird nach einer Zeilenschaltung und einem Zeilenvorschub gesucht.

```
SEARCH( "\0d0a\ " )
SEARCH(data, ifound)
```

Im nachfolgenden Beispiel wird nach dem Wort „alarm“ gesucht:

```
SEARCH( "alarm" )
```

HINWEIS: Bei der Hex-Ersetzung wird mit \0000\ eine Zeichenkette beendet; folglich wird aus „xxx\0000\yyy“ „xxx“.

SET



Mit dem SET-Befehl wird ein mathematischer Ausdruck verarbeitet und ein numerischer Wert (numvar) mit dem Ergebnis der Auswertung aktualisiert.

Bei der Verwendung dieses Befehls gilt Folgendes:

- Es muss eine Ziel-numvar gefolgt von einem Gleichheitszeichen gefolgt von einer beliebigen Kombination von () - + * /, Ziffern und numerischen Variablen.
- Rechts vom Gleichheitszeichen muss mindestens ein numerischer Wert angegeben werden.
- Eingebettete Klammern können in beliebiger Anzahl verwendet werden.
- Sämtliche Argumente werden in eine Float-Variable umgewandelt; das Ergebnis wird in den Typ (Ganzzahl- oder Float-Variable) der Ziel-numvar umgewandelt.
- Nach dem Gleichheitszeichen können bis zu 98 Einträge stehen; zu diesen Einträgen zählen: (,), *, /, +, -, beliebige Ziffern und numerische Variablen.
- Wenn Vorgänge dieselbe Reihenfolge hinsichtlich der Vorgangsebene aufweisen, werden sie von links nach rechts verarbeitet; die Vorgangsreihenfolge geht aus der nachfolgenden Tabelle hervor.

Ebene 1	:	()	Beispiel: Klammer
Ebene 2	:	*/	Beispiel: Multiplikation, Division
Ebene 3	:	+ -	Beispiel: Addition, Subtraktion

Format

```
SET(idest = <expr>) or SET(fdest = <expr>)
```

Hierbei gilt:

```
set_command ::= SET(<idest>=<expr>) | SET(<fdest>=<expr>)
expr ::= (<expr>)
        | expr ( '+' | '-' | '*' | '/' ) expr
        | ivar | fvar | number
```

Datentyp

Argument	Typ	Beschreibung
idest	numvar (AUSGABE)	Die numerische Variable (fvar oder ivar), in der der Wert gespeichert wird.
inum1	Numerisch (EINGABE)	Eine fvar, ivar oder Zahl.
inum2	Numerisch (EINGABE) [OPTIONAL]	Eine fvar, ivar oder Zahl.
inum3	Numerisch (EINGABE) [OPTIONAL]	Eine fvar, ivar oder Zahl.
...	Numerisch (EINGABE) [OPTIONAL]	Eine fvar, ivar oder Zahl.

Beispiel:

```
SET(idest=inum1)
SET(i_loop=10)
SET(idest=inum1+inum2)
SET(idest=(inum1+inum2) * inum3)
SET(i_counter=i_counter+1)
SET(i_val = (ivar)*(ivar/3) + 15/fvar - (5 + 20/iloop))
```

SETBYTES



Mit dem SETBYTES-Befehl können Sie Byte in einer Zeichenkettenvariablen auf einen bestimmten Wert einstellen, der entweder als Ganzzahl oder als Zeichenkette übergeben wurde. Wenn die Übergabe als Ganzzahl erfolgt ist, liegt der gültige Bereich zwischen 0 und 255. Wenn eine Zeichenkette als Austauschparameter verwendet wird, wird die Zeichenkette in der Zielzeichenkettenvariable ab der Indexposition eingefügt.

Format

```
SETBYTES(dest_string, index, replace)
```

Datentypen

Argument	Typ	Beschreibung
dest_string	svar (EINGABE/AUSGABE)	Die Zeichenkettenvariable, in der Byte ersetzt werden.
index	Numerisch (EINGABE)	Der Index (beim Zählen der Byte wird mit 0 für das erste Byte begonnen) für die dest_string, in dem die Byte zum Ersetzen verwendet werden.
replace	Zeichenkette (EINGABE) Oder Ganzzahl (EINGABE)	Die Zeichenkettenbyte, die in die dest_string geschrieben werden. Der Wert, der für das Index-#n-Byte in der Zielzeichenkette festgelegt werden soll.

Beispiel:

```
COPY(string:"Bandwidth Util. = 22%")
SETBYTES(string, 18, "44")
```

Aktueller Inhalt von Ausgabevariablen:

```
string = "Bandwidth Util. = 44%"
```

SETCONFIG



Mit dem -Befehl wird eine Systemeigenschaft festgelegt. Die aktuelle Einstellung für die Systemeigenschaft kann dann mithilfe des [GETCONFIG](#)-Befehls abgerufen werden. Diese Befehle werden zum Festlegen von Systemeigenschaften und zum Abrufen aktueller Werte für Systemeigenschaften verwendet, die sich u. U. regelmäßig ändern, etwa eine Protokolldatei, die jeden Tag in das aktuelle Datum umbenannt wird.

Folgende Systemeigenschaften sind verfügbar:

Systemeigenschaften	Beispiel
▪ System.OS.Family	Solaris und Windows
▪ System.OS.Name	Windows 2000
▪ System.OS.Version.Major	5
▪ System.OS.Version.Minor	0
▪ System.Net.Hostname	ESECServer
▪ System.Net.IP_List	Durch Strichpunkt getrennte Liste der IPs für diesen Host. Beispiel: "172.163.3.45;172.45.2.1"

Ziehen Sie auch den [GETCONFIG](#)-Befehl zurate.

Dieser Befehl umfasst zwei Parameter.

- Mit dem ersten erforderlichen Parameter wird die festzulegende Konfigurationsoption ("FileConnector.InputFile" oder "FileConnector.OutputFile") definiert.
- Mit dem zweiten erforderlichen Parameter wird der festzulegende Konfigurationswert definiert.

Format

```
SETCONFIG(Config Option, Value)
```

Datentypen

Argument	Typ	Beschreibung
Config Option	Zeichenkette (EINGABE)	Name der festzulegenden Konfigurationsvariablen. Eingabedatei = "FileConnector.InputFile" Ausgabedatei = "FileConnector.OutputFile"
Value	Zeichenkette svar (EINGABE)	Konfigurationseinstellung.

Beispiel:

```
SETCONFIG("FileConnector.InputFile", s_inputfilename)  
SETCONFIG("FileConnector.OutputFile", s_outputfilename)
```

Aktueller Inhalt von Ausgabevariablen:

```
"C:\test.dat"
```

SHELL



Mit dem SHELL-Befehl wird ein Shell-Skript oder -Befehl ausgeführt.

Format

```
SHELL(command [, wait_parameter][, wait_return_status])
```

Datentypen

Argument	Typ	Beschreibung
command	Zeichenkette (EINGABE)	Pfad und Dateiname des auszuführenden Befehls. Standardmäßig wird die PATH-Umgebungsvariable verwendet.
wait/no_wait	numvar [OPTIONAL]	Ermöglicht es dem SHELL-Befehl zu warten (bzw. nicht zu warten), bis das gestartete Programm die Ausführung abgeschlossen hat, bevor die Verarbeitung fortgesetzt wird. 0 = Nicht warten 1 = Auf Abschluss des Programms warten
return_status	numvar [OPTIONAL]	Numerischer Wert, wenn die wait/no_wait-Option verwendet wird. ERFOLG = 1 MISSERFOLG = 0

Im nachfolgenden Beispiel wird eine PC-Stapeldatei bzw. ein UNIX-Shell-Skript initiiert:

```
SHELL("device_poll")
```

Im nachfolgenden Beispiel wird Editor gestartet:

```
SHELL("c:\winnt\system32\notepad.exe")
```

Im nachfolgenden Beispiel wird gewartet, bis die Ausführung des clock-Befehls abgeschlossen ist:

```
SHELL("clock",1)
```

Im nachfolgenden Beispiel wird gewartet, bis die Ausführung einer PC-Stapeldatei bzw. eines UNIX-Shell-Skript abgeschlossen ist; anschließend wird der jeweilige Rückgabestatus abgerufen:

```
SHELL("device_poll",1,i_ret)
```

Im nachfolgenden Beispiel wird der clock-Vorgang ausgeführt und nicht auf dessen Abschluss gewartet:

```
SHELL("clock",0)
```

SKIP



Mit dem SKIP-Befehl wird dem Rx Buffer Pointer-Wert eine Zahl hinzugefügt.

Die Zahl kann positiv oder negativ sein. Wenn die resultierende Rx Buffer Pointer Position kleiner als 0 ist, wird der Rx Buffer Pointer auf 0 eingestellt. Wenn die resultierende Rx Buffer Pointer Position über das Ende des Empfangspuffers hinausreicht, wird der Rx Buffer Pointer so eingestellt, dass er auf das letzte Byte im Empfangspuffer verweist.

Format

```
SKIP([+ | -] iskip_amount)
```

Datentypen

Argument	Typ	Beschreibung
iskip_amount	Numerisch (EINGABE)	Die Anzahl der Byte, um die der Rx verschoben werden soll

Beispiel:

```
SKIP(iskip_amount)
SKIP(+iskip_amount)
SKIP(-iskip_amount)
SKIP(5)
SKIP(-1)
```

Aus den nachfolgenden Beispielen geht die Rx Buffer Pointer Position, also die Position des Empfangspufferzeigers, nach dem skip-Befehl für die entsprechenden Daten hervor:

```
aaaaaa bbbbbb c d ee
      ^
```

```
SKIP(-2)
aaaaaa bbbbbb c d ee
      ^
```

```
SKIP(-1)
```

```
aaaaaa bbbbb c d ee
      ^
```

```
SKIP(0)
aaaaaa bbbbb c d ee
      ^
```

```
SKIP(1)
aaaaaa bbbbb c d ee
      ^
```

```
SKIP(2)
aaaaaa bbbbb c d ee
      ^
```

```
SKIP(3)
aaaaaa bbbbb c d ee
      ^
```

```
SKIP(4)
aaaaaa bbbbb c d ee
      ^
```

```
SKIP(8)
aaaaaa bbbbb c d ee
      ^
```

SKIPWORD



Mit dem SKIPWORD-Befehl wird der Rx Buffer Pointer so verändert, dass er auf den Anfang eines Words zeigt.

Für diesen Befehl ist ein Wort eine beliebige Sequenz aufeinanderfolgender druckbarer Byte, die durch mindestens ein nicht druckbares Byte voneinander getrennt werden. Druckbare Byte werden als ASCII- und erweiterte ASCII-0-255-Zeichen definiert (gemäß ISO 8859-1).

Durch die Verwendung positiver und negativer skip-Werte springt der Rx Buffer Pointer vorwärts oder rückwärts durch den Puffer, bis zum ersten bzw. nächsten druckbaren Byte im Empfangspuffer.

Der Rx Buffer Pointer bewegt sich nicht über das Ende des Empfangspuffers hinaus bzw. vor den Anfang des Empfangspuffers, selbst wenn er durch den SKIPWORD-Befehl dazu veranlasst wird.

Bei einem Wert von 0 bleibt der Rx Buffer Pointer unverändert. Der SKIPWORD-Befehl behandelt alle Zeichen unter 33 sowie zwischen 126 und 161 als Leerzeichen.

Format

SKIPWORD([+ | -] iwords)

Datentypen

Argument	Typ	Beschreibung
iwords	Numerisch (EINGABE)	Die Anzahl der Wörter, um die der Rx Buffer Pointer im Empfangspuffer verschoben werden soll.

Beispiel:

```
SKIPWORD(iwords)
SKIPWORD(3)
SKIPWORD(+iwords)
SKIPWORD(-iwords)
SKIPWORD(-4)
```

Aus den nachfolgenden Beispielen geht die Rx Buffer Pointer Position, also die Position des Empfangspufferzeigers, nach dem SKIPWORD-Befehl für die entsprechenden Daten hervor:

```
aaaaaa bbbbb c d ee
      ^
```

```
SKIPWORD(-2)
aaaaaa bbbbb c d ee
      ^
```

```
SKIPWORD(-1)
aaaaaa bbbbb c d ee
      ^
```

```
SKIPWORD(0)
aaaaaa bbbbb c d ee
      ^
```

```
SKIPWORD(1)
aaaaaa bbbbb c d ee
      ^
```

```
SKIPWORD(2)
aaaaaa bbbbb c d ee
      ^
```

```
SKIPWORD(3)
aaaaaa bbbbb c d ee
      ^
```

```

SKIPWORD(4)
aaaaaa bbbbb c d ee
          ^

```

```

SKIPWORD(5)
aaaaaa bbbbb c d ee
          ^

```

SOCKETW



Mit dem SOCKETW-Befehl wird ein BLOCKIERUNGSFREIER (Netzwerk-Byte-STREAM-Socket) Öffnen-, Verbinden-, Schreiben auf Socket-Vorgang (IP- und TCP-Port) durchgeführt und das Socket geschlossen. Optional gibt dieser Befehl den Status des Socket-Schreibversuchs zurück.

Format

```
SOCKETW(address, i_port, data [, istat])
```

Datentypen

Argument	Typ	Beschreibung
address	Zeichenkette (EINGABE)	IP-Adresse des Socket.
i_port	Numerisch (EINGABE)	TCP-Portnummer des Socket.
data	Zeichenkette (EINGABE)	Datenzeichenkette, die auf den Socket geschrieben werden soll.
istat	numvar (AUSGABE)	Optional zurückgegebener Status. istat = Anzahl der geschriebenen Byte; > 0 (ERFOLG) istat = 0 (MISSERFOLG)

Beispiele:

```

SOCKETW("192.168.15.25", 5051, "Data Write Socket")
SOCKETW("192.168.15.25", i_port, "Data to Socket\0d\")
SOCKETW(s_ip_address, i_port, "\54AF0D0B91\ ", i_status)
SOCKETW(s_ip_address, i_port, "\54AF0D0B91\ ", f_status)
SOCKETW(s_ip_address, 6004, "\54AF0D0B91\ ", f_status)
SOCKETW(s_ip_address, 6004, sdata, f_status)

```

STONUM



Mit dem STONUM- (String To Number-)Befehl wird eine Zeichenkettenvariable (svar) in eine numerische Variable (numvar) umgewandelt.

ACHTUNG: Zeichenkettenvariablen, bei denen es sich nicht um die Zeichenkettendarstellung eines Ganzzahl- oder Float-Werts handelt, können zu unvorhersehbaren Ergebnissen führen. Sämtliche Ganzzahlwerte sind auf 2147483647 beschränkt; größere Werte werden auf 2147483647 gekürzt.

Format

```
STONUM(string, ivar)
```

Datentypen

Argument	Typ	Beschreibung
inum	numvar (AUSGABE)	Die numerische Variable, in der die Zahl gespeichert ist (ivar oder fvar).
Zeichenkette	Zeichenkette (EINGABE)	Die Zeichenkettendarstellung einer Zahl (Beispiel: "306").

Beispiel:

```
STONUM(source, idest)
STONUM(string_number, ivar)
STONUM("6512", ivar)
```

STRIP bzw. STRIP-ASCII-RANGE



Mit dem STRIP-Befehl werden sämtliche Vorkommnisse der strip-Zeichenkette bzw. des ASCII-Bereichs aus der svar entfernt. Mit dem STRIP-Befehl werden stets strip-Vorgänge mit mehreren Durchläufen vorgenommen, bis die strip-Zeichenkette bzw. der ASCII-Bereich in der Zielzeichenkettenvariable nicht mehr gefunden wird.

Geben Sie bei der Verwendung dieses Befehls die Zeichenkettenvariable an, aus der Zeichen entfernt werden können. Bei den verbleibenden Parametern kann es sich entweder um eine Zeichenkette oder den Start- und Endwert eines numerischen Bereichs handeln.

HINWEIS: Im visuellen Editor von Collector Builder werden STRIP und STRIP-ASCII-RANGE als separate Befehle aufgelistet. Es handelt sich hierbei um ein- und denselben Befehl. Sie werden als Beschreibungen von Varianten desselben Befehls zur Verfügung gestellt. Wenn Sie im Texteditor STRIP-ASCII-RANGE verwenden möchten, geben Sie STRIP ein.

Format

```
STRIP(dest, strip)
STRIP(dest, start ASCII range, stop ASCII range)
```


Datentypen

Argument	Typ	Beschreibung
dest	svar (EINGABE/AUSGABE)	Die Zeichenkettenvariable, die die Zeichenkettenvariablen enthält, aus denen abhängig vom zweiten Argument Byte entfernt werden.
strip bzw. start ASCII range	Zeichenkette oder numerisch (EINGABE)	Die Zeichenkette bzw. der start ASCII-Wert, der aus der Zielzeichenkette entfernt werden soll.
stop ASCII range	Numerisch (EINGABE [optional])	stop ASCII-Wert HINWEIS: Wenn „start ASCII range“ angegeben wird, ist dieser Parameter erforderlich.

Bei den nachfolgenden Beispielen handelt es sich um strip-Vorgänge mit mehreren Durchläufen.

```
COPY(test:"THEHELLOE")
STRIP(test, "HELLO")
```

Nach dem STRIP()-Befehl weist die test-Variable den Wert THE auf.

```
COPY(test2:"ABCDDEDDDFGDDH")
STRIP(test2, "D")
```

Nach dem STRIP()-Befehl weist die test2-Variable den Wert ABCEFGH auf.

```
COPY(test3:"ABCDDEDDDFGDDH")
STRIP(test3, 68, 69)
```

Nach dem STRIP()-Befehl weist die test3-Variable den Wert ABCFGH auf.

TBOSETCOMMAND



Mit dem TBOSETCOMMAND-Befehl wird ein 3-Byte-TBOS-Befehlspaket erstellt, das mithilfe des TBOS-(Telemetry Byte Oriented Serial-)Protokolls an ein Gerät übermittelt werden kann.

TBOS-Anzeigenummer, Befehlsnummer und Befehlstyp werden allesamt verwendet, um das richtige TBOS-Befehlspaket (3-Byte-Paket) in die Ausgabezeichenkettenvariable zu schreiben. Das Format des mit diesem Parsing-Befehl erstellten TBOS-Pakets wird in den nachfolgenden Tabellen für Remote-Befehlsanforderungen erläutert.

Zeichen 1		
Bitzahl	Wert	Bedeutung
8	0	Betriebscode:
7	1	01 = Remote-Befehlsanforderung (Zeichen 1)

Zeichen 1		
Bitzahl	Wert	Bedeutung
6 5 4	MSB LSB	Anzeigenummer: 000 = Nr. 1 001 = Nr. 2 ... 111 = Nr. 7
3	0	Keine Bedeutung
2 1	MSB LSB	Typ: 00 = momentary 01 = latch 10 = unlatch

Zeichen 2		
Bitzahl	Wert	Bedeutung
8 7	1 0	Betriebscode: 10 = Remote-Befehlsanforderung (Zeichen 2)
6 5 4 3 2 1	MSB LSB	Remote-Befehlsnummer: 000000 = Nr. 1 000001 = Nr. 2 ... 111111 = Nr. 63

Zeichen 3		
Bitzahl	Wert	Bedeutung
8 7 6 5 4 3 2 1	1 1 0 0 1 1 0 0	Echo von Zeichen: Diese Remote-Befehlsantwort ist das Echo dieses Byte an den Port zurück.

Format

```
TBOSSETCOMMAND(cmd_bytes, idisp_num, icmd_num, type)
```

Datentypen

Argument	Typ	Beschreibung
cmd_bytes	svar (AUSGABE)	Die Hex-Datenbyte (3 Byte insgesamt), die in diese Zeichenkettenvariable geschrieben werden und mit denen im Feld für die Übertragung des nächsten Status die Übertragung an ein TBOS-Gerät möglich ist.
idisp_num	Numerisch (EINGABE)	Die TBOS-Anzeigenummer (bzw. Adresse) des Geräts(1–8). HINWEIS: Der gültige Bereich für „idisp_num“ ist auf 1–8 beschränkt; bei der Verwendung eines anderen Werts wird die Ausgabe (cmd_bytes) ausschließlich auf Nullen eingestellt, "\00 00 00\".
i_cmd_num	Numerisch (EINGABE)	Die TBOS-Befehlsnummer (1–64). HINWEIS: Der gültige Bereich für „i_cmd_num“ ist auf 1–64 beschränkt; bei der Verwendung eines anderen Werts wird die Ausgabe (cmd_bytes) ausschließlich auf Nullen eingestellt, "\00 00 00\".
type	Numerisch (EINGABE) Oder Zeichenkette (EINGABE)	Der TBOS-Befehlstyp (0–2): 0 = momentary 1 = latch 2 = unlatch <hr/> HINWEIS: Der gültige Bereich für „type“ ist auf 0–2 beschränkt; bei der Verwendung eines anderen Werts wird „type“ standardmäßig auf „0 = “momentary“ eingestellt. <hr/> Der TBOS-Befehlstyp im Zeichenkettenformat. "momentary" oder "m" = Vorübergehend "latch" oder "l" = Verriegeln "unlatch" oder "u" = entriegeln Bei dieser Zeichenkette muss die Groß-/Kleinschreibung nicht beachtet werden.

Beispiel:

```
TBOSSETCOMMAND(string_cmd_bytes, 1, 1, 0)
TBOSSETCOMMAND(s_bytes, 1, 1, "latch")
TBOSSETCOMMAND(s_bytes, i_display, i_cmd_num, "U")
TBOSSETCOMMAND(s_bytes, i_display, i_cmd_num, 2)
TBOSSETCOMMAND(s_bytes, 1, 1, "momentary")
TBOSSETCOMMAND(s_bytes, 1, 1, "latch")
```

Stellen Sie in jedem Fall sicher, dass „output cmd_bytes“ auf "\00 00 00\" eingestellt ist, damit auf Fehler hinsichtlich außerhalb des Bereichs liegender Eingaben geprüft werden kann. Beispiel:

```
TBOSETCOMMAND(cmd_bytes, i_display, i_cmd_num, "M")
IF(cmd_bytes = "\00 00 00\") /* INPUTS OUT OF RANGE */
...
ENDIF( )
```

Im nachfolgenden Beispiel wird ein TBOS-Befehl für Anzeigenummer 5, Befehlsnummer 33, vom Typ „unlatched“ erstellt.

```
TBOSETCOMMAND(sbytes, 5, 33, 2)
```

Aktueller Inhalt von Ausgabevariablen:

```
sbytes = "\ba0 cc\"
```

TBOSETREQUEST



Mit dem TBOSETREQUEST- Befehl wird ein 1-Byte-TBOS-Anforderungspaket erstellt, das mithilfe des TBOS-(Telemetry Byte Oriented Serial-)Protokolls an ein Gerät übermittelt werden kann. Die TBOS-Anzeige- und Anforderungsnummer wird verwendet, um das richtige TBOS-Absucheanforderungsbyte in die Ausgabezeichenkettenvariable zu schreiben. Das Format des mit diesem Parsing-Befehl erstellten TBOS-Pakets wird in den nachfolgenden Tabellen für Zeichenabsucheanforderungen und -antworten erläutert.

Zeichen 1 – Zeichenabsucheanforderung		
Bitzahl	Wert	Bedeutung
8	0	Betriebscode:
7	0	00 = Zeichenabsucheanforderung
6	MSB	Anzeigenr.:
5		000 = Nr. 1
4	LSB	001 = Nr. 2
		...
		111 = Nr. 3
3	MSB	Typ:
2		000 = Nr. 1
1	LSB	001 = Nr. 2
		...
		111 = Nr. 8

Zeichen 1 – Zeichenabsucheantwort		
Bitzahl	Wert	Bedeutung
8	MSB	Jedes Bit in diesem Antwortbyte hat eine besondere Bedeutung, basierend auf der gesendeten Zeichenzahl (1–8) und dem Protokoll des Geräts der gesendeten Anzeigenummer (1–8).
7		
6		
5		
4		
3		
2		
1	LSB	

Format

```
TBOSSETREQUEST(cmd_bytes, idisp_num, irequest_num)
```

Datentypen

Argument	Typ	Beschreibung
cmd_bytes	svar (AUSGABE)	Das Hex-Datenbyte wird in diese Zeichenkettenvariable geschrieben und kann im Feld für die Übertragung des nächsten Status für die Übertragung an ein TBOS-Gerät verwendet werden.
idisp_num	Numerisch (EINGABE)	Die TBOS-Anzeigenummer (bzw. Adresse) des Geräts(1–8). HINWEIS: Der gültige Bereich für „idisp_num“ ist auf 1–8 beschränkt; bei einem anderen Wert wird die Ausgabe, cmd_bytes, ausschließlich auf Nullen eingestellt, "\00\".
irequest_num	Numerisch (EINGABE)	Die TBOS-Absuchezeichenzahl (1–8). HINWEIS: Der gültige Bereich für „irequest_num“ ist auf 1–8 beschränkt; bei einem anderen Wert wird die Ausgabe, cmd_bytes, auf Nullen eingestellt, "\00\".

Beispiel:

```
TBOSSETREQUEST(string_request_byte, 1, 1)
TBOSSETREQUEST(s_byte, idisp_num, i_scan_number)
```

Im nachfolgenden Beispiel wird ein TBOS-Absuchanforderungszeichen für Anzeigenummer 2 und Anforderungsnummer 1 erstellt.

```
TBOSSETREQUEST(sbytes, 2, 1)
```

Aktueller Inhalt von Ausgabevariablen:

```
sbytes = "\08\"
```

TIME



Mit dem TIME-Befehl wird die aktuelle Uhrzeit (im Format HH-MM-SS) in eine Zeichenkettenvariable, ivar oder fvar, kopiert.

Format

`TIME(dest)`

Datentypen

Argument	Typ	Beschreibung
dest	svar (AUSGABE)	Die Zeichenkettendarstellung der Uhrzeit wird in diese Zeichenkettenvariable geschrieben (z. B. „23-11-55“).
	numvar (AUSGABE)	Die Anzahl der Sekunden seit 00:00:00 UTC (Universal Coordinated Time) des 1. Januar 1970 wird in diese numerische Variable (kann eine fvar sein) geschrieben.

Beispiel:

`TIME(time_of_day)`

`TIME(i_num_seconds)`

`TIME(f_num_seconds)`

HINWEIS: Wenn Sie eine fvar verwenden, ist die zurückgegebene Zeit auf die Mikrosekunde genau.

TOKENIZE



Mit dem TOKENIZE- Befehl wird jede Komponente einer Zeichenkette zwischen den Begrenzungszeichen in ein Zeichenkettenarray kopiert. Dies kann hilfreich sein, wenn Sie durch bestimmte Zeichen begrenzte Daten aus einer Datei lesen und Daten an ein Skript übermitteln, das nach Bedarf ausgeführt wird.

Jedes Zeichen in der Zeichenkette wird als potenzielles Token-Trennzeichen behandelt. Wenn beispielsweise das Token-Trennzeichen „THE END“ verwendet wird, wird nicht die gesamte Zeichenkette als Trennzeichen verwendet. Stattdessen kommen einzelne Zeichen als potenzielle Trennzeichen zum Einsatz:

" T "

" H "

" E "

" E "

" N "

" D "

Format

```
TOKENIZE(data, delimiter, tokens[], itokens)
```

Datentypen

Argument	Typ	Beschreibung
data	svar (EINGABE)	Die Daten, die mit Token versehen werden sollen (Beispiel: „xterm subres 33 50“).
delimiter	Zeichenkette (EINGABE)	Die Begrenzungszeichen, durch die die Token voneinander getrennt werden sollen.
token	array (AUSGABE)	Das Array mit Token, wie in den mit Begrenzungszeichen versehenen Zeichenketteneingabedaten angegeben.
itokens	numvar (AUSGABE)	Die Anzahl der Token, die in das Token-Zeichenkettenarray aufgenommen werden.

Beispiel:

```
COPY(data:"This|Data|Is|Tokenized")
TOKENIZE(data, "|",tokens[], inumtokens)
```

Aktueller Inhalt von Ausgabevariablen:

```
inumtokens = 4
tokens[0]= "This"
tokens[1]= "Data"
tokens[2]= "Is"
tokens[3]= "Tokenized"
```

Im nachfolgenden Beispiel werden folgende Daten an das Skript übergeben:

```
"There#are|several*fields|in*this#string".
```

Diese drei unterschiedlichen Token-Trennzeichen werden zur Verwendung empfohlen: #, | und *.

Aktueller Inhalt von Ausgabevariablen:

```
i_tokens = 7
messages[0] = "There"
messages[1] = "are"
messages[2] = "several"
messages[3] = "fields"
messages[4] = "in"
messages[5] = "this"
messages[6] = "string"
```

Im nachfolgenden Beispiel lauten die Daten im Empfangspuffer wie folgt:

```
"Firewall Alarm - Major;Denial of Service Alarm - Major;"  
COPY(rxbuff:)  
TOKENIZE(rxbuff,";",msgs[],i_msgs)
```

Aktueller Inhalt von Ausgabevariablen:

```
i_msgs = 2  
msgs[0] = "Firewall Alarm - Major"  
msgs[1] = "Denial of Service Alarm - Major"
```

TOLOWER



Mit dem TOLOWER- Befehl wird der Inhalt einer Zeichenkettenvariable ausschließlich in Kleinbuchstaben umgewandelt. Der Inhalt der Zeichenkettenvariable, die durch diese Befehl übergeben wird, wird in ausschließlich Kleinbuchstaben umgewandelt.

Format

```
TOLOWER(stringvar)
```

Datentypen

Argument	Typ	Beschreibung
stringvar	Zeichenkette (EINGABE/AUSGABE)	Die Zeichenkettenvariable, die die Zeichenkette enthält, die in ausschließlich Kleinbuchstaben umgewandelt werden soll.

Beispiel:

```
s_var = "This Is Lower Case"  
TOLOWER(s_var)
```

Ergebnis:

```
s_var = "this is lower case"
```

TOUPPER



Mit dem TOUPPER-Befehl wird der Inhalt einer Zeichenkettenvariable ausschließlich in Großbuchstaben umgewandelt. Der Inhalt der Zeichenkettenvariable, die durch diese Befehl übergeben wird, wird in ausschließlich Großbuchstaben umgewandelt.

Format

```
TOUPPER(stringvar)
```


Datentypen

Argument	Typ	Beschreibung
stringvar	Zeichenkette (EINGABE/AUSGABE)	Die Zeichenkettenvariable, die die Zeichenkette enthält, die in ausschließlich Großbuchstaben umgewandelt werden soll.

Beispiel:

```
s_var = "This Is Upper Case"
toupper(s_var)
```

Ergebnis:

```
s_var = "THIS IS UPPER CASE"
```

TRANSLATE



Mit dem TRANSLATE- Befehl wird eine Datei mit kommagetrennten Werten (.csv-Datei) in den Arbeitsspeicher geladen; auf diese Weise kann schnell ermittelt werden, ob der Schlüsseleintrag in der Datei enthalten ist, und andere mit dem Schlüssel verknüpfte Daten können abgerufen werden.

Folgende Elemente stehen mit dem TRANSLATE-Befehl in Zusammenhang.

- Kommagetrennter Wert (Comma-Separated Value, CSV)
- Schlüsselsuchvorgänge ohne Beachtung der Groß-/Kleinschreibung
- Found-Status
- Datenvariablen

Datei mit kommagetrennten Werten (CSV-Datei)

Die CSV-Datei stellt einen relativen Pfad von einem Skriptverzeichnis des Collectors dar. Die Bearbeitung dieser Dateien wird von Collector Builder nicht unterstützt; folglich empfiehlt Novell deren Erstellung in Microsoft Excel. Beim Dateinamen kann es sich um eine Zeichenkette oder eine Variable handeln.

Das .csv-Dateiformat wird im nachfolgenden Beispiel anhand einer Datei namens friends.csv verdeutlicht:

```
key1,data1,data2,data3
Bob,blue,25,210
Alice,green,19,110
Pat,purple,36,145
```

Zur Suche nach einem bestimmten Freund in Ihrer friend.csv-Datei sieht der TRANSLATE-Befehl folgendermaßen aus:

```
TRANSLATE("Bob", "friends.csv", i_found)
```

Oder

```
COPY(s_Name: "Bob" )
TRANSLATE(s_Name, "friends.csv", i_found)
```

Schlüsselsuchvorgänge ohne Beachtung der Groß-/Kleinschreibung

Beim Schlüsselparameter kann es sich entweder um eine Zeichenkette oder eine Zeichenkettenvariable handeln. Zudem wird eine Ganzzahl bzw. Variable unterstützt. Wenn die .csv-Datei in den Arbeitsspeicher geladen wird, wird der Schlüssel der einzelnen Einträge auf Kleinschreibung eingestellt. Der Schlüssel im TRANSLATE-Befehl wird ebenfalls intern auf die Kleinschreibung eingestellt, um Schlüsselsuchvorgänge ohne Beachtung der Groß-/Kleinschreibung zu ermöglichen.

Fortsetzung des Beispiels für eine .csv-Datei:

```
TRANSLATE("boB", "friends.csv", i_found)
```

Auch hiermit wird Bob in der .csv-Datei gefunden.

Found-Status

Der Found-Status wird auf 1 eingestellt, wenn der Schlüssel in der .csv-Datei enthalten ist; er wird auf 0 eingestellt, wenn der Schlüssel nicht in der .csv-Datei enthalten ist. Eine .csv-Datei, die nur aus Schlüsseleinträgen besteht, kann mit dem TRANSLATE-Befehl verwendet werden, um lediglich zu ermitteln, ob der Schlüssel Mitglied dieser Datei ist. Aus Sicherheitsgründen enthält eine .csv-Datei u. U. eine Liste bekannter feindlicher IP-Adressen bzw. gültiger Benutzernamen mit weiteren Richtlinieninformationen, etwa Berechtigungen und zulässige Zugriffszeiten.

HINWEIS: Schlüssel, die Bereiche ausdrücken, werden nicht unterstützt: IP-Adresse und numerische Bereiche.

Datenvariablen

Neben der Feststellung, ob ein Schlüsseleintrag in der .csv-Datei enthalten ist oder nicht, können verknüpfte Daten für diesen Schlüssel abgerufen werden. Eine variable Anzahl an Skriptvariablen kann verwendet werden, um anzugeben, in welchen Variablen die Daten gespeichert werden sollen. Zeichenketten-, Ganzzahl- und Float-Variablen werden unterstützt. Sämtliche Dateneinträge werden als Zeichenketten gespeichert und in den Variablentyp umgewandelt, der im TRANSLATE-Befehl angegeben ist.

Fortsetzung des Beispiels für friends.csv:

```
Bob,blue,25,210
Alice,green,19,110
Pat,purple,36,145
```

Die verknüpften Daten können hiermit abgerufen werden:

```
TRANSLATE(s_friend, "friends.csv", i_found, s_color,
          i_age, i_weight)
```

Hierbei gilt:

- Wenn s_friend „Alice“ enthält, entspricht i_found 1, s_color entspricht green, i_age entspricht 19 und i_weight entspricht 110.
- Wenn der Schlüsseleintrag nicht gefunden wird, bleiben die Variablen unverändert (s_color, i_age, i_weight).
- Angenommen, der Eintrag für „Alice“ lautet wie folgt:
Alice,green,19,

In diesem Fall wird bei Verwendung desselben TRANSLATE die Variable `i_weight` gelöscht (0 für Ganzzahlen, 0.0 für Float-Werte und ""-Zeichenketten). `s_color` lautet hier `green` und `i_age` lautet 19.

- Angenommen, der Eintrag für „Alice“ lautet wie folgt:
`Alice,green,,thin,Ford`

In diesem Fall wird bei Verwendung desselben TRANSLATE die Variable `i_age` gelöscht, `thin` wird in eine Ganzzahl umgewandelt (0) und in `i_weight` geschrieben. `s_color` lautet hier `green` und `Ford` wird hier ignoriert.

- Angenommen, der Eintrag für „Alice“ lautet wie folgt:
`Alice,25,19,110`

In diesem Fall enthält bei Verwendung desselben TRANSLATE die Variable `s_color` 25. `i_age` lautet hier 19 und `i_weight` lautet 110.

Format

```
TRANSLATE(<key>, <csv_file>, <found_status> [,
        <variable>, ...])
```

Datentypen

Argument	Typ	Beschreibung
key		Der Schlüssel, nach dem in der .csv-Datei gesucht werden soll.
csv_file		Der Dateiname der .csv-Datei.
found_status		Die Ganzzahlvariable, die auf 1 eingestellt wird, wenn der Schlüssel in der .csv-Datei gefunden wird; sie wird auf 0 eingestellt, wenn der Schlüssel in der .csv-Datei nicht gefunden wird.
Variable		Die Liste der Variablen, in die die mit dem Schlüssel verknüpften Daten geschrieben werden sollen.

TRIM



Entfernt alle Leerzeichen (Leerschritte) von beiden Enden einer Zeichenkette und ersetzt mehrere Leerzeichen in einer Zeichenkette durch einzelne Leerzeichen. Zu den Leerzeichen zählen folgende Zeichen:

- `<tab>` (Tabulator)
- `<carriage-return>` (Zeilenschaltung)
- `<newline>` (Neue Zeile)
- `<vertical-tab>` (Vertikaltabulator)
- `<form-feed>` (Formularvorschub)
- `<space>` (Leerschritt)

Format

```
TRIM(svar)
```

Datentypen

Argument	Typ	Beschreibung
Zeichenkette	svar (EINGABE)	Zeichenkette, aus der Leerzeichen entfernt werden soll. Die resultierende Zeichenkette wird in der Ausgabevariablen gespeichert.

Beispiel:

```
COPY(s_var: " Hello      World  ")
TRIM(s_var)
```

Aktueller Inhalt von Ausgabevariablen:

```
s_var = " Hello World "
```

WHILE



Mit dem WHILE-Befehl kann der Steuerungsfluss als Schleife gestaltet werden.

Für den While-Befehl gilt Folgendes:

- Wenn das Ergebnis der WHILE()-Anweisung „true“ ist, werden die Parsing-Befehle nach dem WHILE() bis zum nächsten ENDWHILE() ausgeführt.
- Wenn das Ergebnis des WHILE()-Vergleichs „false“ ist, werden keine Parsing-Befehle zwischen WHILE() und ENDWHILE() ausgeführt.

Obwohl sämtliche Datentypen auf beiden Seiten des Operators der WHILE()-Anweisung zulässig sind, können numerische Werte nur mit numerischen Werten und Zeichenketten nur mit Zeichenketten verglichen werden.

Mögliche Operatoren für den WHILE()-Vergleich: <, =, >, <=, >=, <>, &, + oder ^.

ACHTUNG: Verwenden Sie den logischen NOT-Operator (^) nicht gemeinsam mit einer Zeichenkettenvariablen. Anderenfalls tritt ein Syntaxfehler auf.

Der direkte Vergleich mit einer negativen Zahl ist nicht möglich. Sie haben folgende Möglichkeiten:

- COMPARE der Parsing-Funktion verwenden
- Indirekten Vergleich wie folgt durchführen:

```
SET(i_compare_val=-10)
WHILE(ivar >i_compare_val)
SET(ivar=ivar-1)
ENDWHILE()
```

Format

```
WHILE(<expr>)
```

Hierbei gilt:

```
expr ::= var
      | (<expr>)
      | ^ <expr>
```

Hierbei muss <expr> zu einer Ganzzahl- oder Float-Variablen werden.

| <expr> <|=|>|<=|>=|<>|&|+ <expr>

Hierbei müssen die beiden <expr> zum selben Typ werden.

Datentypen

Argument	Typ	Beschreibung
data1	Beliebig (EINGABE)	Die Daten, die mit „data2“ verglichen werden sollen. Wenn „data2“ nicht verwendet wird, wird daraus ein logischer Operator (0 = false, Rest = true).
logical operator	< = > <= >= <> & + ^	Kleiner Gleich Größer Kleiner oder gleich Größer oder gleich Nicht gleich Logischer AND-Operator Logischer OR-Operator Logischer NOT-Operator
data2	Beliebig (EINGABE) [OPTIONAL]	Die Daten, die mit „data1“ verglichen werden sollen. Müssen denselben Typ wie „data1“ aufweisen.
...	Wie oben	Mit bis zu 200 einzelnen Parametern können komplexe logische Ausdrücke erstellt werden.

Beispiel:

```
WHILE(i<3)
SET(i=i+1)
ALERT("Still in loop")
ENDWHILE()
ALERT("Exited loop")
```


4

Wizard-Administratorfunktionen

HINWEIS: Die Begriffe „Agent“ und „Collector“ sind austauschbar. Im Folgenden werden Agenten als „Collectors“ bezeichnet.

Dieses Kapitel richtet sich an den Wizard-Systemadministrator. Hier werden unterschiedliche administrative Funktionen des Systemadministrators erläutert und Informationen zu den Hintergrundvorgängen des Wizard aufgeführt.

HINWEIS: Bei der ersten Ausführung von Wizard Collector Builder wird unter Umständen eine Meldung mit etwa folgendem Wortlaut ausgegeben: "Verzeichnis 'Collectors' ist nicht vorhanden." Es wird automatisch erstellt. Einige Informationen sind möglicherweise nicht mehr verfügbar". Wenn Sie auf "OK" klicken, wird das Verzeichnis erstellt und Wizard Collector Builder wird gestartet. Wenn diese Meldung nach der erstmaligen Ausführung von Collector Builder angezeigt wird, wurde das Collector-Verzeichnis möglicherweise versehentlich gelöscht und Sie müssen überprüfen, ob Informationen nicht mehr verfügbar sind.

Wizard-Dienstprogramme und -Anwendungen

Wizard besteht aus einer Benutzeroberfläche (Collector Builder) und mehreren zusätzlichen Dienstprogrammen, die gemeinsam mit Collector Builder für die Netzwerküberwachung sorgen.

Collector Builder

Collector Builder ist die Wizard-Benutzeroberfläche. Über Collector Builder können Sie die Collectors in Ihrem Netzwerk sowie die Ports und Skripts konfigurieren, die zur Kommunikation mit den Hosts verwendet werden. Collector Builder kann nur unter Windows ausgeführt werden.

HINWEIS: Wenn nach der Neupositionierung von Wizard-Fenstern ein Anzeigeproblem auftritt, überprüfen Sie die Anzeigeeinstellungen über die Microsoft Windows-Systemsteuerung. Deaktivieren Sie auf der Registerkarte „Effekte“ das Kontrollkästchen „Fensterinhalt beim Ziehen anzeigen“.

Port

In Wizard; über Ports kann ein Collector die Sicherheitsereignisdaten im Netzwerk suchen, indem die IP-Adresse und andere Informationen zur Quelle (Sicherheitsgerät [Router, IDS, Switch usw.]) bereitgestellt werden. Mit jeder Zeile in der Portkonfigurations-Tabelle wird ein Collector-Skript für eine Ereignisquelle ausgeführt.

Collector Manager

Über Collector Manager wird die Portverarbeitung gestartet und gestoppt.

Collector Engine

Über Collector Engine verfolgt die Verarbeitung der Schablonenlogik für die einzelnen Ports. Für jeden aktiven Port wird eine Collector Engine ausgeführt.

popup.exe

Das popup.exe-Dienstprogramm wird von Collector Engine zur Verarbeitung von Popup- und Anzeige-Parsing-Befehlen herangezogen.

popup.cfg

Bei der popup.cfg-Datei handelt es sich um eine optionale Datei, mit der die Zeitüberschreitung von Popup- und Anzeige-Parsing-Befehlen gesteuert wird. Wenn Ihnen keine popup.cfg zur Verfügung steht, kommt es zu keiner Zeitüberschreitung der Anzeige- und Popup-Parsing-Befehle.

Geben Sie zur Festlegung eines Zeitüberschreitungswerts für den Anzeige-Befehl folgende Anweisung ein:

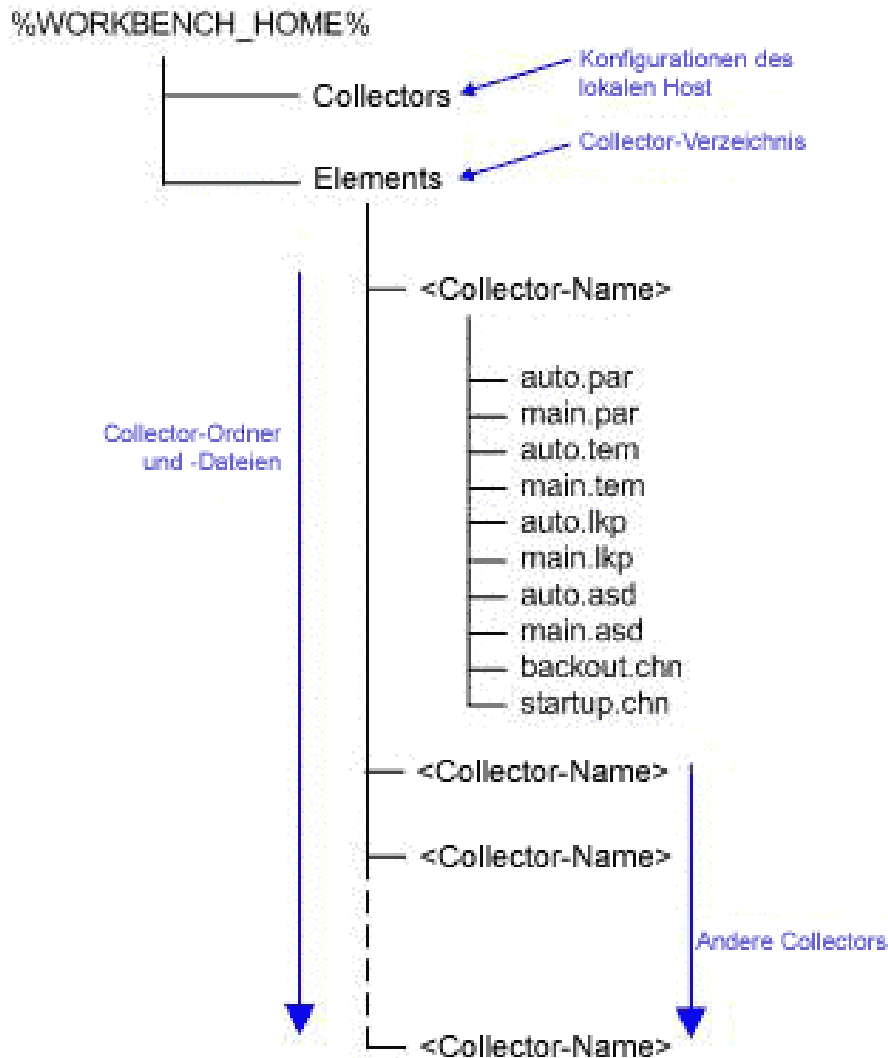
```
displaytimeout <true/false>.
```

Die Zeitüberschreitung für die Anzeige ist auf 20 Sekunden eingestellt.

Geben Sie zur Festlegung eines Zeitüberschreitungswerts für den Popup-Befehl folgende Anweisung ein:

```
timeout <timeout in seconds>.
```


Wizard-Verzeichnisstruktur



Schlüssel

Collectors	Portkonfigurationsdateien (Wizard-Hosts)
Elements	Collector-Dateien
.par	Parameterdateien
.tem	Schablonendateien
.lkp	Suchdateien
.asd	Beschreibungsdateien für aktiven Status
backout.chn	Zurücksetzungsskriptdateien
startup.chn	Startskriptdateien

5

META-Tags für Wizard und Sentinel

HINWEIS: Die Begriffe „Agent“ und „Collector“ sind austauschbar. Im Folgenden werden Agenten als „Collectors“ bezeichnet.

HINWEIS: Bei MS SQL 2000 darf die Ereignisgröße 8 KB nicht überschreiten.

In META-Tags werden META-Daten gespeichert. Als META-Daten werden Informationen zu Daten bzw. vordefinierte Variablennamen für META-Daten bezeichnet. Beispielsweise wird die Quellen-IP eines Angriffs im META-Tag „SourceIP“ gespeichert. Produktnamen werden im META-Tag „ProductName“ gespeichert. Daten zum Auffüllen von META-Tags werden entweder aus den Geräteprotokolldaten extrahiert oder als Teil der Collector-Verarbeitung festgelegt.

Wenn Sie auf die Ereigniskonfiguration und die Zuordnungsfunktion in Sentinel Data Manager zugreifen möchten, klicken Sie auf die Registerkarte für die Ereignisse.

HINWEIS: In der FreeForm-RuleLg-Sprache für Korrelationsregeln gilt Folgendes: Wenn einer Kennung ein „e.“ vorangestellt ist, beispielsweise „e.crt“, dient dies als Verweis auf aktuelle Ereignisse. Wenn einer Kennung ein „w.“ vorangestellt ist, beispielsweise „w.crt“, dient dies als Verweis auf Verlaufsereignisse.

Der Wert in der Spalte für die Collector-Variable ist der Name der Collector-Variablen, die zum Auffüllen des entsprechenden META-Tags festgelegt werden muss. Weitere Informationen zu Parsing-Befehlen finden Sie in Kapitel 3 und der spezifischen Collector-Dokumentation unter

```
%ESEC_HOME%\wizard\elements\<Collector name>\docs.
```

HINWEIS: In der nachfolgenden Tabelle werden Kennungen und META-Tags in Sentinel Control Center verwendet. Collector-Variablen kommen beim Collector-Parsing zum Einsatz. Nicht alle META-Tags weisen eine zugehörige Collector-Variable auf.

Die in der Typ-Spalte angegebenen Typen weisen folgende Eigenschaften auf:

- Zeichenkette – Auf 255 Zeichen beschränkt (wenn nicht anders angegeben)
- Ganzzahl – 32-Bit-Ganzzahl mit Vorzeichen
- UUID – Hexadezimal-Zeichen mit 36 Zeichen (mit Bindestrichen) bzw. 32 Zeichen (ohne Bindestriche) im Format XXXXXXXX-XXXX-XXXX-XXXX-XXXXXXXXXXXX (z. B. – 6A5349DA-7CBF-1028-9795-000BCDFFF482)
- Datum – Für die Collector-Variable muss das Datum als Anzahl der Millisekunden seit dem 1. Januar 1970, 00:00:00 GMT (Greenwich Mean Time) angegeben werden. Bei der Anzeige in Sentinel Control Center werden META-Tags vom Typ „Datum“ in einem normalen Datumsformat angezeigt.
- IPv4 – IP-Adresse in punktierter Dezimalnotation (z. B. – xxx.xxx.xxx.xxx)

Kennung	META-Tag	Typ	Beschreibung	Collector-Variable
CorrelatedEventUuids	ceu	Zeichenkette	Liste mit Ereignis-UUIDs, die mit diesem korrelierten Ereignis verknüpft sind. Nur für korrelierte Ereignisse relevant.	
Criticality	crt	Ganzzahl	Die Gefährlichkeit des in diesem Ereignis identifizierten Bestands.	s_CRIT
Ct1 thru Ct2 (Reserved Customer)	ct1 thru ct2	Zeichenkette	Reserviert für die Verwendung durch Kunden für kundenspezifische Daten (Zeichenkette)	s_CT1 und s_CT2
Ct3 (Reserved Customer)	ct3	Ganzzahl	Reserviert für die Verwendung durch Kunden für kundenspezifische Daten (Zahl)	s_CT3
CustomerVar1 thru CustomerVar10	cv1 thru cv10	Ganzzahl	Reserviert für die Verwendung durch Kunden für kundenspezifische Daten (Zahl)	s_CV1 bis s_CV10
CustomerVar11 thru CustomerVar20	cv11 thru cv20	Datum	Reserviert für die Verwendung durch Kunden für kundenspezifische Daten (Datum)	s_CV11 bis s_CV20
CustomerVar21 thru CustomerVar29	cv21 thru cv29	Zeichenkette	Reserviert für die Verwendung durch Kunden für kundenspezifische Daten (Zeichenkette)	s_CV21 bis s_CV29
CustomerVar30 thru CustomerVar34	cv30 thru cv34	Zeichenkette	Reserviert für die Verwendung durch Kunden für kundenspezifische Daten (Zeichenkette) Kann Zeichenkettenlängen von bis zu 4000 Zeichen verarbeiten.	s_CV30 bis s_CV34
CustomerVar35 thru CustomerVar89	cv35 thru cv89	Zeichenkette	Reserviert für die Verwendung durch Kunden für kundenspezifische Daten (Zeichenkette)	s_CV35 bis s_CV89
SARBOX	cv90	Zeichenkette	Spezifische Sarbanes Oxley-Daten.	s_CV90
HIPAA	cv91	Zeichenkette	Spezifische Health Insurance Portability and Accountability Act-(HIPAA-)Daten.	s_CV91
GLBA	cv92	Zeichenkette	Spezifische Gramm-Leach-Bliley Act-(GLBA-)Daten.	s_CV92
FISMA	cv93	Zeichenkette	Spezifische Federal Information Security Management Act-(FISMA-)Daten.	s_CV93
NISPOM	cv94	Zeichenkette	Spezifische National Industrial Security Program Operating Manual-(NISPOM-)Daten.	s_CV94

Kennung	META-Tag	Typ	Beschreibung	Collector-Variable
SIPCountry	cv95	Zeichenkette	Ursprungsland-IP.	s_CV95
DIPCountry	cv96	Zeichenkette	Zielland-IP.	s_CV96
CustomerVar97 thru CustomerVar100	cv97 thru cv100	Zeichenkette	Reserviert für die Verwendung durch Kunden für kundenspezifische Daten (Zeichenkette)	s_CV97 bis s_CV100
DateTime	dt	Datum	Datum und Uhrzeit (normalisiert) des Ereignisses, wie vom Collector angegeben.	
DestinationHostName	dhn	Zeichenkette	Der Name des Ziel-Hosts, an den das Ereignis gerichtet war.	s_DHN
DestinationIP	dip	IPv4	Die IP-Zieladresse, an die das Ereignis gerichtet war.	s_DIP
DestinationPort	dp	Zeichenkette (32)	Der Ziel-Port, an den das Ereignis gerichtet war.	s_DP
DestinationUserName	dun	Zeichenkette	Der Zielbenutzername, für den versucht wurde, eine Aktion durchzuführen. Beispiel: Versuche, das Passwort von „root“ zurückzusetzen.	s_DUN
EventID	id	UUID	Eindeutige Kennung für dieses Ereignis.	
EventTime	et	Zeichenkette	Der normalisierte Uhrzeit des Ereignisses, wie vom Sensor gemeldet; Analysierung in das Format: Y-M-D-H:M:S~AMPM24~TZ.	s_ET
EventName	evt	Zeichenkette	Der beschreibende Name des Ereignisses, wie vom Sensor gemeldet (oder angegeben). Beispiel: „Port Scan“ (Port-Absuchvorgang).	s_EVT
ExtendedInformation	ei	Zeichenkette (1000)	Speichert zusätzliche im Collector erfasste Informationen. Werte innerhalb dieser Variable werden durch Semikolon (;) getrennt. Beispiel: Eine Domäne für eine ID oder Dateinamen.	s_EI
FileName	fn	Zeichenkette (1000)	Der Name des ausgeführten Programms oder der Datei, auf die zugegriffen wurde bzw. die geändert oder beeinträchtigt wurde. Beispiel: Der Name einer infizierten Datei oder eines infizierten Programms, die bzw. das von IDS ermittelt wurde.	s_FN

Kennung	META-Tag	Typ	Beschreibung	Collector-Variable
Message	msg	Zeichenkette (4000)	Der FreeForm-Meldungstext (ohne Formatvorgabe) für das Ereignis.	s_BM
Protocol	prot	Zeichenkette	Das Netzwerkprotokoll des Ereignisses.	s_P
ProductName	pn	Zeichenkette	Gibt den Typ, Hersteller und den Produktcodenamen des Sensors an, von dem das Ereignis generiert wurde. Beispiel: Check Point FireWall=CPFW.	s_PN
ReporterName	rn	Zeichenkette	Der Hostname oder die IP-Adresse des Geräts, in dem ein Ereignis protokolliert wurde oder von dem eine Benachrichtigung hinsichtlich des Ereignisses gesendet wird.	s_RN
ReservedVar1 thru ReservedVar10	rv1 thru rv10	Ganzzahl	Von Novell für Erweiterungen reserviert (Zahl).	s_RV1 bis s_RV10
ReservedVar11 thru ReservedVar20	rv11 thru rv20	Datum	Von Novell für Erweiterungen reserviert (Datum).	s_RV11 bis s_RV20
ReservedVar21 thru ReservedVar25	rv21 thru rv25	UUID	Von Novell für Erweiterungen reserviert (UUID).	s_RV21 bis s_RV25
ControlPack	rv26	Zeichenkette	Sentinel- Steuerungskategorisierungsstufe 1	s_RV26
ControlMonitor	rv27	Zeichenkette	Sentinel- Steuerungskategorisierungsstufe 2	s_RV27
ReservedVar28	rv28	Zeichenkette	Von Novell für Erweiterungen reserviert (Zeichenkette).	s_RV28
SourceIPCountry	rv29	Zeichenkette	Ursprungsland-IP-Adresse.	s_RV29
AttackID	rv30	Zeichenkette	Normalisierte Angriffs-ID (Advisor-Angriffs-ID)	s_RV30
DeviceName	rv31	Zeichenkette	Name des Sicherheitsgeräts	s_RV31
DeviceCategory	rv32	Zeichenkette	Gerätekategorie (AV, DB, ESEC, FW, IDS, OS). AV: Virenschutz DB: Datenbank ESEC: Systemereignis FW: Firewall IDS: Intrusion Detection OS: Betriebssystem	s_RV32
EventContext	rv33	Zeichenkette	Ereigniskontext (Bedrohungsgrad).	s_RV33
SourceThreatLevel	rv34	Zeichenkette	Quellbedrohungsgrad.	s_RV34

Kennung	META-Tag	Typ	Beschreibung	Collector-Variable
SourceUserContext	rv35	Zeichenkette	Quellbenutzerkontext.	s_RV35
DataContext	rv36	Zeichenkette	Datenkontext.	s_RV36
SourceFunction	rv37	Zeichenkette	Quellfunktion.	s_RV37
SourceOperationalContext	rv38	Zeichenkette	Quellbetriebskontext.	s_RV38
MSSPCustomerName	rv39	Zeichenkette	MSSP-Kundenname.	s_RV39
ReservedVar40 thru ReservedVar43	rv40 thru rv43	Zeichenkette	Von Novell für Erweiterungen reserviert (Zeichenkette).	s_RV40 bis s_RV43
DestinationThreatLevel	rv44	Zeichenkette	Zielbedrohungsgrad.	s_RV44
DestinationUserContext	rv45	Zeichenkette	Zielbenutzerkontext.	s_RV45
VirusStatus	rv46	Zeichenkette	Virusstatus.	s_RV46
DestinationFunction	rv47	Zeichenkette	Zielfunktion.	s_RV47
DestinationOperationalContext	rv48	Zeichenkette	Zielbetriebskontext.	s_RV48
ReservedVar49	rv49	Zeichenkette	Von Novell für Erweiterungen reserviert (Zeichenkette).	s_RV49
eSecTaxonomyLevel1	rv50	Zeichenkette	Sentinel- Ereigniscodekategorisierung – Stufe 1.	s_RV50
eSecTaxonomyLevel2	rv51	Zeichenkette	Sentinel- Ereigniscodekategorisierung – Stufe 2.	s_RV51
eSecTaxonomyLevel3	rv52	Zeichenkette	Sentinel- Ereigniscodekategorisierung – Stufe 3.	s_RV52
eSecTaxonomyLevel4	rv53	Zeichenkette	Sentinel- Ereigniscodekategorisierung – Stufe 4.	s_RV53
ReservedVar54 thru ReservedVar55	rv54 thru rv55	Zeichenkette	Von Novell für Erweiterungen reserviert (Zeichenkette).	s_RV54 bis s_RV55
SourceAssetName	rv56	Zeichenkette	Quelle (Bestandsverwaltung) – Bestandsname	s_RV56
SourceMacAddress	rv57	Zeichenkette	Quelle (Bestandsverwaltung) – MAC-Adresse	s_RV57
SourceNetworkIdentity	rv58	Zeichenkette	Quelle (Bestandsverwaltung) – Netzwerkidentität	s_RV58
SourceAssetCategory	rv59	Zeichenkette	Quelle (Bestandsverwaltung) – Bestandskategorie	s_RV59
SourceEnvironmentIdentity	rv60	Zeichenkette	Quelle (Bestandsverwaltung) – Umgebungsidentität	s_RV60
SourceAssetValue	rv61	Zeichenkette	Quelle (Bestandsverwaltung) – Bestandswert	s_RV61
SourceCriticality	rv62	Zeichenkette	Quelle (Bestandsverwaltung) – Gefährlichkeit	s_RV62
SourceSensitivity	rv63	Zeichenkette	Quelle (Bestandsverwaltung) – Vertraulichkeit	s_RV63
SourceBuilding	rv64	Zeichenkette	Quelle (Bestandsverwaltung) – Gebäude	s_RV64

Kennung	META-Tag	Typ	Beschreibung	Collector-Variable
SourceRoom	rv65	Zeichenkette	Quelle (Bestandsverwaltung) – Zimmer	s_RV65
SourceRackNumber	rv66	Zeichenkette	Quelle (Bestandsverwaltung) – Regalnummer	s_RV66
SourceCity	rv67	Zeichenkette	Quelle (Bestandsverwaltung) – Stadt	s_RV67
SourceState	rv68	Zeichenkette	Quelle (Bestandsverwaltung) – Bundesland	s_RV68
SourceCountry	rv69	Zeichenkette	Quelle (Bestandsverwaltung) – Land	s_RV69
SourceZipCode	rv70	Zeichenkette	Quelle (Bestandsverwaltung) – Postleitzahl	s_RV70
SourceAssetOwner	rv71	Zeichenkette	Quelle (Bestandsverwaltung) – Bestandseigentümer	s_RV71
SourceAssetMaintainer	rv72	Zeichenkette	Quelle (Bestandsverwaltung) – Bestandsverwalter	s_RV72
SourceBusinessUnit	rv73	Zeichenkette	Quelle (Bestandsverwaltung) – Unternehmenseinheit	s_RV73
SourceLineOfBusiness	rv74	Zeichenkette	Quelle (Bestandsverwaltung) – Sparte	s_RV74
SourceDivision	rv75	Zeichenkette	Quelle (Bestandsverwaltung) – Division	s_RV75
SourceDepartment	rv76	Zeichenkette	Quelle (Bestandsverwaltung) – Abteilung	s_RV76
SourceAssetId	rv77	Zeichenkette	Quelle (Bestandsverwaltung) – Quellbestands-ID	s_RV77
DestinationAssetName	rv78	Zeichenkette	Ziel (Bestandsverwaltung) – Bestandsname	s_RV78
DestinationMacAddress	rv79	Zeichenkette	Ziel (Bestandsverwaltung) – MAC-Adresse	s_RV79
DestinationNetworkIdentity	rv80	Zeichenkette	Ziel (Bestandsverwaltung) – Netzwerkidentität	s_RV80
DestinationAssetCategory	rv81	Zeichenkette	Ziel (Bestandsverwaltung) – Bestandskategorie	s_RV81
DestinationEnvironmentIdentity	rv82	Zeichenkette	Ziel (Bestandsverwaltung) – Umgebungsidentität	s_RV82
DestinationAssetValue	rv83	Zeichenkette	Ziel (Bestandsverwaltung) – Bestandswert	s_RV83
DestinationCriticality	rv84	Zeichenkette	Ziel (Bestandsverwaltung) – Gefährlichkeit	s_RV84
DestinationSensitivity	rv85	Zeichenkette	Ziel (Bestandsverwaltung) – Vertraulichkeit	s_RV85
DestinationBuilding	rv86	Zeichenkette	Ziel (Bestandsverwaltung) – Gebäude	s_RV86
DestinationRoom	rv87	Zeichenkette	Ziel (Bestandsverwaltung) – Zimmer	s_RV87
DestinationRackNumber	rv88	Zeichenkette	Ziel (Bestandsverwaltung) – Regalnummer	s_RV88

Kennung	META-Tag	Typ	Beschreibung	Collector-Variable
DestinationCity	rv89	Zeichenkette	Ziel (Bestandsverwaltung) – Stadt	s_RV89
DestinationState	rv90	Zeichenkette	Ziel (Bestandsverwaltung) – Bundesland	s_RV90
DestinationCountry	rv91	Zeichenkette	Ziel (Bestandsverwaltung) – Land	s_RV91
DestinationZipCode	rv92	Zeichenkette	Ziel (Bestandsverwaltung) – Postleitzahl	s_RV92
DestinationAssetOwner	rv93	Zeichenkette	Ziel (Bestandsverwaltung) – Bestandseigentümer	s_RV93
DestinationAssetMaintainer	rv94	Zeichenkette	Ziel (Bestandsverwaltung) – Bestandsverwalter	s_RV94
DestinationBusinessUnit	rv95	Zeichenkette	Ziel (Bestandsverwaltung) – Unternehmenseinheit	s_RV95
DestinationLineOfBusiness	rv96	Zeichenkette	Ziel (Bestandsverwaltung) – Sparte	s_RV96
DestinationDivision	rv97	Zeichenkette	Ziel (Bestandsverwaltung) – Division	s_RV97
DestinationDepartment	rv98	Zeichenkette	Ziel (Bestandsverwaltung) – Abteilung	s_RV98
DestinationAssetId	rv99	Zeichenkette	Ziel (Bestandsverwaltung) – Zielbestands-ID	s_RV99
ReservedVar100	rv100	Zeichenkette	Von Novell für Erweiterungen reserviert (Zeichenkette).	s_RV100
Resource	res	Zeichenkette	Der Ressourcename.	s_Res
DeviceAttackName	rt1	Zeichenkette	Zur Verwendung mit Advisor. Angriffsname vom Sicherheitsgerät.	s_RT1
Rt2	rt2	Zeichenkette	Wird mit dem Namen der Korrelationsregel aufgefüllt, wenn eine Korrelationsregel ein Ereignis erstellt.	s_RT2
Rt3	rt3	Ganzzahl	Von Novell für Erweiterungen reserviert (Zahl).	s_RT3
SourceHostName	shn	Zeichenkette	Der Quell-Hostname, von dem das Ereignis ausging.	s_SHN
SourceID	src	UUID	Die eindeutige Kennung für den Sentinel-Vorgang, der dieses Ereignis generiert hat.	
SourceIP	sip	IPv4	Die Quell-IP-Adresse, von der das Ereignis ausging.	s_SIP
SensorName	sn	Zeichenkette	Der Name des "höchsten Detektors" des Ereignisses, wenn als unverarbeitete Daten empfangen. Beispiel: „FW1“ für eine Firewall.	s_SN
Severity	sev	Ganzzahl	Der normalisierte Schweregrad des Ereignisses (0–5).	i_Severity

Kennung	META-Tag	Typ	Beschreibung	Collector-Variable
SourcePort	sp	Zeichenkette (32)	Der Quell-Port, von dem das Ereignis ausging.	s_SP
SensorType	st	Zeichenkette (5)	Der Bezeichner für den Sensortyp, der mit einem einzelnen Buchstaben angegeben wird (N, H, I, O, P, V, C, W). C: Korrelation H: Host-basiert I: Intern (Systemereignis) N: Netzwerk-basiert O: Sonstiges P: Leistung (Systemereignis) V: Virenschutz W: Beobachtungsliste	s_ST
SourceUserName	sun	Zeichenkette	Der Quellenbenutzername, der zum Initiieren des Ereignisses verwendet wurde. Beispiel: "jdoe" während eines „su“-Versuchs.	s_SUN
SubResource	sres	Zeichenkette	Der Name der Teilressource.	s_SubRes
Vulnerability	vul	Ganzzahl	Die Anfälligkeit des in diesem Ereignis identifizierten Bestands.	s_VULN
WizardAgent	agent	Zeichenkette (64)	Sentinel-Collector, der das Ereignis erstellt hat. Bei Systemereignissen handelt es sich hierbei entweder um einen leistungsbezogenen oder einen internen Collector.	
WizardPort	port	Zeichenkette (64)	Portbeschreibung für Sentinel-Collector.	

6

Benutzerberechtigungen für Sentinel Control Center

HINWEIS: Die Begriffe „Agent“ und „Collector“ sind austauschbar. Im Folgenden werden Agenten als „Collectors“ bezeichnet.

Benutzerberechtigungen sind wie folgt unterteilt:

- [Allgemein](#)
 - [Öffentliche Filter](#)
 - [Private Filter](#)
 - [Integrationsaktionen](#)
- [Active View](#)
 - [Menüelemente](#)
 - [Zusammenfassungsansichten](#)
- [iTRAC](#)
 - [Schablonenverwaltung](#)
 - [Vorgangsverwaltung](#)
- [Vorfälle](#)
- [Collector-Verwaltung](#)
- [Analyse](#)
- [Advisor](#)
- [Administration](#)
 - [Korrelation](#)
 - [DAS-Statistik](#)
 - [Ereignisdatei-Info](#)
 - [Serveransichten](#)
 - [Globale Filter](#)
 - [iTRAC-Funktionsverwaltung](#)
 - [Menükonfiguration](#)
 - [Benutzerverwaltung](#)
 - [Benutzersitzungsverwaltung](#)

Standardbenutzer

Das Installationsprogramm erstellt folgende Standardbenutzer in Sentinel Server:

Oracle- und MS SQL-Authentifizierung:

- esecdba – Schemaeigentümer (zum Zeitpunkt der Installation zu konfigurieren).
- esecadm – Sentinel-Administratorbenutzer (zum Zeitpunkt der Installation zu konfigurieren).

HINWEIS: Unter UNIX erstellt das Installationsprogramm zudem den Betriebssystembenutzer mit demselben Benutzernamen und Passwort.

- esecrpt – Reporter-Benutzer, Passwort wie admin-Benutzer.
- ESEC_CORR – Correlation Engine-Benutzer, Verwendung zur Erstellung von Vorfällen.
- esecapp – Sentinel-Anwendungsbenutzername zum Herstellen der Datenbankverbindung.

Windows-Authentifizierung:

- Sentinel-DB-Administrator – Schemaeigentümer (zum Zeitpunkt der Installation zu konfigurieren).
- Sentinel-Administrator – Sentinel-Administratorbenutzer (zum Zeitpunkt der Installation zu konfigurieren).
- Sentinel Report-Benutzer – Reporter-Benutzer, Passwort wie admin-Benutzer.
- Sentinel-DB-Anwendungsbenutzer – Sentinel-Anwendungsbenutzername zum Herstellen der Datenbankverbindung.

Allgemein

Berechtigungsname	Beschreibung
Arbeitsbereich speichern	Ermöglicht dem Benutzer das Speichern von Einstellungen. Wenn diese Berechtigung nicht verfügbar ist, wird der Benutzer beim Abmelden bzw. beim Beenden von Sentinel Control Center nie zum Speichern der an den Einstellungen vorgenommenen Änderungen aufgefordert.
Spaltenverwaltung	Ermöglicht dem Benutzer die Verwaltung der Spalten in den Active Views-Tabellen.
Snapshot	Ermöglicht dem Benutzer die Erstellung eines Snapshot von Active Views-Tabellen.

Allgemein – Öffentliche Filter

Berechtigungsname	Beschreibung
Öffentliche Filter erstellen	Ermöglicht dem Benutzer die Erstellung eines Filters mit der Eigentümer-ID ÖFFENTLICH zu erstellen. Wenn dieser Benutzer nicht über diese Berechtigung verfügt, wird der Wert ÖFFENTLICH nicht als eine der Eigentümer-IDs aufgeführt, für die der Benutzer einen Filter erstellen kann.
Öffentliche Filter ändern	Ermöglicht es dem Benutzer, einen öffentlichen Filter zu ändern.
Öffentliche Filter löschen	Ermöglicht es dem Benutzer, einen öffentlichen Filter zu löschen.

Allgemein – Private Filter

Berechtigungsname	Beschreibung
Private Filter erstellen	Ermöglicht es dem Benutzer, private Filter für sich selbst oder andere Benutzer zu erstellen.
Private Filter ändern	Ermöglicht es dem Benutzer, eigene private Filter sowie von anderen Benutzern erstellte private Filter zu ändern.
Private Filter löschen	Ermöglicht es dem Benutzer, eigene private Filter sowie von anderen Benutzern erstellte private Filter zu löschen.
Private Filter anzeigen/verwenden	Ermöglicht es dem Benutzer, eigene private Filter sowie von anderen Benutzern erstellte private Filter anzuzeigen.

Allgemein – Integrationsaktionen

Berechtigungsname	Beschreibung
An HP Open View senden	Ermöglicht es dem Benutzer, Ereignisse, Vorfälle und zugehörige Objekte an HP-OVO zu senden.
Ereignis an HP Service Desk senden	Ermöglicht es dem Benutzer, Ereignisse, Vorfälle und zugehörige Objekte an HP Service Desk zu senden.
An Remedy Help Desk senden	Ermöglicht es dem Benutzer, Ereignisse, Vorfälle und zugehörige Objekte an Remedy zu senden.

Active Views

Berechtigungsname	Beschreibung
Active Views-Registerkarte anzeigen	Ermöglicht es dem Benutzer, die Active Views-Registerkarte, das Active Views-Menü sowie andere entsprechende Funktionen im Zusammenhang mit der Active Views-Registerkarte anzuzeigen und zu verwenden.

Active Views – Menüelemente

Berechtigungsname	Beschreibung
Zugewiesene Menüelemente verwenden	Ermöglicht es dem Benutzer, in der Ereignis-Tabelle von Active Views (dem Kontextmenü) zugewiesene Menüelemente zu verwenden.
Zu vorhandenem Vorfall hinzufügen	Ermöglicht es dem Benutzer, in der Ereignis-Tabelle von Active Views (dem Kontextmenü) Ereignisse zu vorhandenen Vorfällen hinzuzufügen.
Aus Vorfall entfernen	Ermöglicht es dem Benutzer, in der Ereignis-Tabelle von Active Views (dem Kontextmenü) Ereignisse aus einem vorhandenen Vorfall zu entfernen.
Ereignisse mailen	Ermöglicht es dem Benutzer, über die Ereignis-Tabelle von Active Views (das Kontextmenü) Ereignisse per Email zu versenden.
Advisor-Angriffsdaten anzeigen	Ermöglicht es dem Benutzer, den Advisor-Angriffsdatenstrom anzuzeigen.
Anfälligkeit anzeigen	Ermöglicht es dem Benutzer, die Ausgabe eines Nessus-Absuchvorgangs anzuzeigen.

Active Views – Zusammenfassungsansichten

Berechtigungsname	Beschreibung
Zusammenfassungsansichten verwenden/anzeigen	Ermöglicht es dem Benutzer, auf die Active Views-Diagramme zuzugreifen.

iTRAC

Berechtigungsname	Beschreibung
iTRAC-Registerkarte anzeigen	Ermöglicht es dem Benutzer, die iTRAC-Registerkarte, das Active Views-Menü sowie andere entsprechende Funktionen im Zusammenhang mit der iTRAC-Registerkarte anzuzeigen und zu verwenden.
Aktivitätsverwaltung	Ermöglicht es dem Benutzer, auf die Aktivitätsverwaltung zuzugreifen.

Schablonenverwaltung

Berechtigungsname	Beschreibung
Schablonen-Manager anzeigen/verwenden	Ermöglicht es dem Benutzer, auf den Schablonen-Manager zuzugreifen.
Schablonen erstellen/ändern	Ermöglicht es dem Benutzer, Schablonen zu erstellen und zu ändern.

Vorgangsverwaltung

Berechtigungsname	Beschreibung
Vorgangs-Manager anzeigen/verwenden	Ermöglicht es dem Benutzer, auf den Vorgangsansicht-Manager zuzugreifen.
Vorgänge steuern	Ermöglicht es dem Benutzer, den Vorgangsansicht-Manager zu verwenden.

Vorfälle

Berechtigungsname	Beschreibung
Registerkarte „Vorfälle anzeigen“	Ermöglicht es dem Benutzer, die Registerkarte „Vorfälle anzeigen“, das zugehörige Menü sowie andere entsprechende Funktionen im Zusammenhang mit der Registerkarte „Vorfälle anzeigen“ anzuzeigen und zu verwenden.
Incident Administration	Ermöglicht es dem Benutzer, einen Vorfall zu ändern.
Vorfälle anzeigen	Ermöglicht es dem Benutzer, die Details eines Vorfalls anzuzeigen. Wenn der Benutzer über diese Berechtigung nicht verfügt, wird das Fenster mit den Vorfallsdetails nicht angezeigt, wenn der Benutzer im Navigator-Fenster oder auf der Vorfall-Registerkarte des jeweiligen Falls auf einen Vorfall doppelklickt.
Vorfälle erstellen	Ermöglicht es dem Benutzer, Vorfälle im Ereignismenü zu erstellen, auf das durch Rechtsklick auf ein Ereignis zugegriffen werden kann.
Vorfälle ändern	Ermöglicht es dem Benutzer, einen Vorfall mit Fenster mit den Vorfallsdetails zu ändern.
Vorfälle löschen	Ermöglicht es dem Benutzer, Vorfälle zu löschen.
Vorfälle zuweisen	Ermöglicht es dem Benutzer, einen Vorfall im Fenster zum Ändern und Erstellen von Vorfällen zuzuweisen.
Vorfälle mailen	Ermöglicht es dem Benutzer, interessante Vorfälle per Email zu senden.
Vorfallsaktionen	Ermöglicht es dem Benutzer, die Konfiguration/Ausführung von Vorfallsaktionen zu aktivieren/zu deaktivieren.

Collector-Verwaltung

Berechtigungsname	Beschreibung
Collectors anzeigen	<ul style="list-style-type: none"> Collectors-Registerkarte in Sentinel Control Center anzeigen Registerkarte mit Wizard-Hosts in Collector Builder anzeigen
Collectors steuern	<ul style="list-style-type: none"> Umfasst dieselben Fähigkeiten wie die Berechtigung „Collectors anzeigen“ Ermöglicht die Steuerung von Collectors über Sentinel Control Center Ermöglicht die Steuerung von Collectors über Wizard Collector Builder

Berechtigungsname	Beschreibung
Collector-Administration	<ul style="list-style-type: none"> ▪ Umfasst dieselben Fähigkeiten wie die Berechtigung „Collectors steuern“ ▪ In Collector Builder: Bearbeiten und Bereitstellen von Collectors ▪ In Collector Builder: Erstellen, Bearbeiten und Kompilieren von Collectors sowie Durchführen der Fehlersuche für Collectors ▪ In Collector Builder: Herauf- und Herunterladen von Collectors ▪ In Collector Builder: Exportieren von Wizard-Hosts ▪ In Collector Builder: Hinzufügen, Bearbeiten und Löschen von Ports ▪ In Collector Builder: Festlegen von Portoptionen

Die Steuerung umfasst folgende Schritte:

- Einzelne Ports starten/stoppen
- Alle Ports starten/stoppen
- Hosts neu starten
- Hosts umbenennen

Analyse

Berechtigungsname	Beschreibung
Registerkarte „Analyse anzeigen“	Ermöglicht es dem Benutzer, die Registerkarte „Analyse anzeigen“, das zugehörige Menü sowie andere entsprechende Funktionen im Zusammenhang mit der Registerkarte „Systemübersicht anzeigen“ anzuzeigen und zu verwenden.

Advisor

Berechtigungsname	Beschreibung
Registerkarte „Advisor anzeigen“	Ermöglicht es dem Benutzer, die Registerkarte „Advisor anzeigen“, das zugehörige Menü sowie andere entsprechende Funktionen im Zusammenhang mit der Registerkarte „Advisor anzeigen“ anzuzeigen und zu verwenden.

Verwaltung

Berechtigungsname	Beschreibung
Registerkarte „Verwaltung anzeigen“	Ermöglicht es dem Benutzer, die Registerkarte „Verwaltung anzeigen“, das zugehörige Menü sowie andere entsprechende Funktionen im Zusammenhang mit der Registerkarte „Verwaltung anzeigen“ anzuzeigen und zu verwenden.

Verwaltung – Korrelation

Berechtigungsname	Beschreibung
Correlation Engine-Manager verwenden/anzeigen	Ermöglicht es dem Benutzer, Correlation Engine anzuzeigen und zu verwenden.
Korrelationsregeln verwenden/anzeigen	Ermöglicht es dem Benutzer, die Korrelationsregeln zu starten und zu stoppen.

Verwaltung – Globale Filter

Berechtigungsname	Beschreibung
Globale Filter anzeigen/verwenden	Ermöglicht es dem Benutzer, auf das Fenster „Globale Filterkonfiguration“ zuzugreifen.
Globale Filter ändern	Ermöglicht es dem Benutzer, die globale Filterkonfiguration zu ändern. <div style="border: 1px solid black; padding: 5px; margin-top: 10px;">HINWEIS: Für den Zugriff auf diese Funktion muss die Berechtigung zum Anzeigen globaler Filter ebenfalls zugewiesen werden.</div>

Verwaltung – Menükonfiguration

Berechtigungsname	Beschreibung
Menükonfiguration	Ermöglicht es dem Benutzer, auf das Fenster „Menükonfiguration“ zuzugreifen und neue Optionen hinzufügen, die im Ereignismenü angezeigt werden, wenn Sie mit der rechten Maustaste auf ein Ereignis klicken.

Verwaltung – DAS-Statistik

Berechtigungsname	Beschreibung
DAS-Statistik	Ermöglicht es dem Benutzer, die DAS-Aktivität (binäre DAS-Aktivität und DAS-Abfrageaktivität) anzuzeigen.

Verwaltung – Ereignisdatei-Info

Berechtigungsname	Beschreibung
Ereignisdatei-Info	Ermöglicht es dem Benutzer, den Ereignisdateistatus anzuzeigen.

Verwaltung – Serveransichten

Berechtigungsname	Beschreibung
Server anzeigen	Ermöglicht es dem Benutzer, den Status sämtlicher Vorgänge zu überwachen.
Server steuern	Ermöglicht es dem Benutzer, Vorgänge zu starten, neu zu starten und zu stoppen.

Verwaltung – Benutzerverwaltung

Berechtigungsname	Beschreibung
Benutzerkonto verwenden/anzeigen	Ermöglicht es dem Benutzer, Benutzerkonten zu verwenden und anzuzeigen.

Berechtigungsname	Beschreibung
Benutzerkonto erstellen	<p>Ermöglicht es dem Benutzer, ein Benutzerkonto zu erstellen.</p> <hr/> <p>HINWEIS: Für den Zugriff auf diese Funktion muss die Berechtigung zum Anzeigen von Benutzerkonten ebenfalls zugewiesen werden.</p>
Bestehendes Benutzerkonto ändern	<p>Ermöglicht es dem Benutzer, ein bestehendes Benutzerkonto zu ändern.</p> <hr/> <p>HINWEIS: Für den Zugriff auf diese Funktion muss die Berechtigung zum Anzeigen von Benutzerkonten ebenfalls zugewiesen werden.</p>
Benutzerkonto löschen	<p>Ermöglicht es dem Benutzer, ein bestehendes Benutzerkonto zu löschen.</p> <hr/> <p>HINWEIS: Für den Zugriff auf diese Funktion muss die Berechtigung zum Anzeigen von Benutzerkonten ebenfalls zugewiesen werden.</p>

Verwaltung – Benutzersitzungsverwaltung

Berechtigungsname	Beschreibung
Benutzersitzungsverwaltung	Ermöglicht es dem Benutzer, aktive Benutzervorgänge (Anmeldungen bei Sentinel Control Center) anzuzeigen, zu sperren und zu beenden.

Verwaltung – iTRAC-Funktionsverwaltung

Berechtigungsname	Beschreibung
iTRAC-Funktionsverwaltung	Ermöglicht die Verwendung und Anzeige des Funktionsmanagers auf der Admin-Registerkarte.

7

Sentinel Correlation Engine

HINWEIS: Die Begriffe „Agent“ und „Collector“ sind austauschbar. Im Folgenden werden Agenten als „Collectors“ bezeichnet.

Bei Sentinel Correlation Engine handelt es sich um eine speicherresidente Multi-Thread-Anwendung. Durch Multi-Threading kann Correlation Engine die Vorteile der Multiprozessorhardware, z. B. von SMP-Maschinen (Symmetric Multiprocessing), nutzen.

Correlation Engine ist für den Empfang von Daten von Sicherheits- und Netzwerkgeräten und anderen Anwendungsquellen konzipiert. Mit der Engine wird normalerweise innerhalb bestimmter Zeiträume nach signifikanten Mustern gesucht. Diese Muster könnten Hinweise auf Angriffe, Eingriffe, Fehler oder Missbrauch darstellen. Beim Erstellen eines korrelierten Ereignisses wird im Feld „rt2“ der Name der Korrelationsregel eingefügt.

Sentinel Correlation Engine ermöglicht eine skalierbare Bereitstellung. Mithilfe dieser Architektur ist eine Bereitstellung eines verteilten Netzwerks von Correlation Engine-Instanzen möglich, die zusammenarbeiten, um in Echtzeit sicherheitsrelevante Daten zu korrelieren. Dazu zählen Sicherheitsereignisse, die in Echtzeit überwacht werden, Ergebnisse zu Anfälligkeitsprüfungen für potenzielle Zielsysteme und Bestandsinformationen, mit denen die relative Wichtigkeit für geschäftskritische Prozesse und deren Verbindung mit anderen Systemen in der Organisation angegeben wird.

Sentinel Correlation Engine ist regelbasiert. Die Verarbeitung der Correlation Engine-Instanz kann über Regeln gesteuert werden, die im Regeleditor des Sentinel Control Center erstellt wurden. Der Regeleditor basiert auf einer Sammlung von Regelassistenten, die mehrere Optionen zum Erstellen von Regeln bereitstellen. Zu den Regelassistenten zählen:

- [Beobachtungsliste](#)
- [Allgemeine Korrelation](#)
- [Erweiterte Korrelation](#)
- [FreeForm RuleLg](#)

Korrelationsfiltertypen

Für „Beobachtungsliste“, „Grundlegende Korrelation“ und „Erweiterte Korrelation“ kann aus vier unterschiedlichen Filtertypen ausgewählt werden. Dazu gehören:

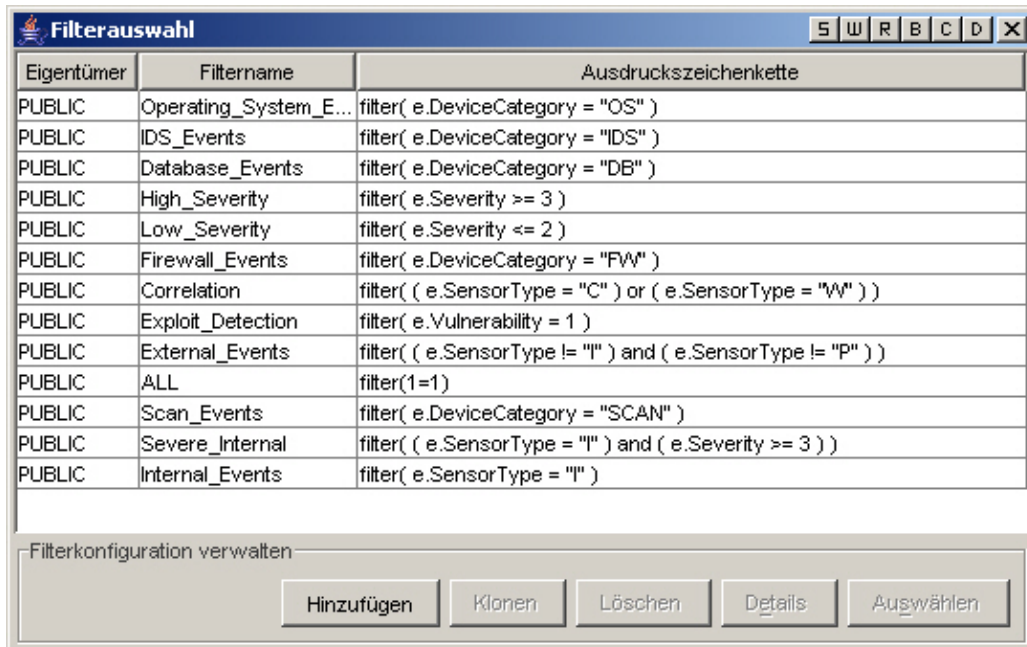
- Alle zulassen - Äquivalent zum Ausführen eines Filtergrads größer oder gleich null (0).
- Muster – Alle regulären Ausdrücke mit einer Grep-Syntax. Eine Regel kann nach einer spezifischen Quellen-IP-Adresse eines Hackers suchen und Sie jedes Mal benachrichtigen, wenn diese IP-Adresse in Ereignisnachrichten ermittelt wird.
- Filtermanager – Eine Dropdown-Liste, die den Filtermanager anzeigt, um einen neuen Filter auszuwählen oder zu erstellen.
- Editor – Erstellen von Kriterien zum Einschließen oder Ausschließen von Ereignissen auf Grundlage boolescher Algebra.

Mustertypen-Korrelationsfilter

Ein Mustertypen-Korrelationsfilter verwendet alle regulären Ausdrücke mit einer Grep-Syntax. Die Übereinstimmung des regulären Ausdrucks erfolgt über eine Verkettung aller verfügbaren META-Tags für die einzelnen eingehenden Ereignisse. Beispielsweise kann mit virusXYZ in den verfügbaren META-Tags für ein einzelnes eingehendes Ereignis nach einer Zeichenkette virusXYZ gesucht werden.

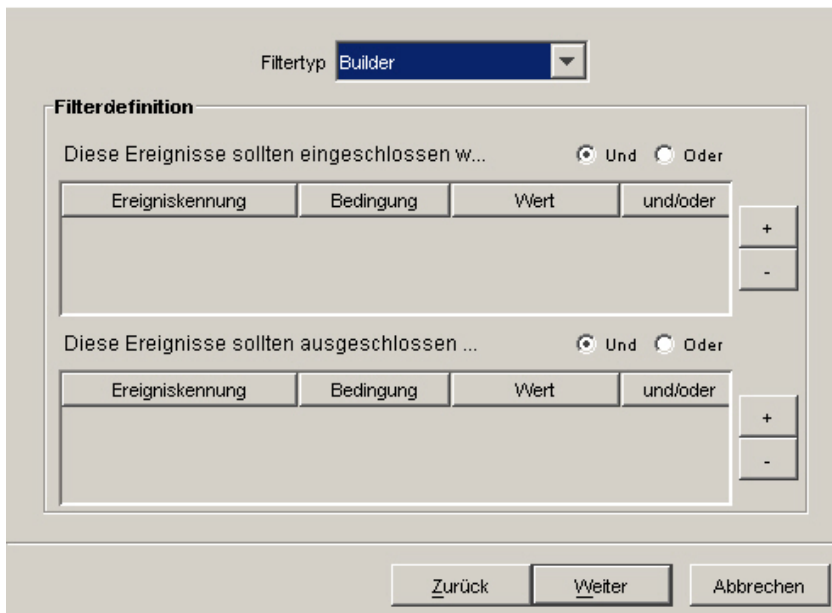
Filtermanager-Typ-Korrelationsfilter

Mithilfe dieser Option können Sie im Fenster „Filtermanager“ einen vorhandenen Filter auswählen oder einen Filter erstellen, den Sie in Ihrer Korrelation verwenden möchten.



Editortyp-Korrelationsfilter

Es gibt zwei Bestandteile für den Editortyp-Korrelationsfilter. Der eine Bestandteil sind die Einschlusskriterien (welche Ereignisse sollen in den Musterabgleich eingeschlossen werden), der andere die Ausschlusskriterien (welche Ereignisse sollen aus dem Musterabgleich ausgeschlossen werden).



- Welche Ereignisse sollen im Musterabgleich enthalten sein – Geben Sie mithilfe dieser Tabelle die Bedingungen an, um Einschränkungen festzulegen, welche Ereignisse die Korrelation auslösen.
 - Ereigniskennung – Die Spalte „Ereigniskennung“ ist eine Dropdown-Liste, in der verfügbare Ereignis-Tags (auch als META-Tags bezeichnet) enthalten sind, für die eine Korrelation erfolgen kann.
 - Bedingung – Die Spalte „Bedingung“ ist eine Dropdown-Liste mit Operatoren, die zum Erstellen einer Korrelationsbedingung verwendet werden können.
 - Wert – Bei der Spalte „Wert“ handelt es sich um ein FreeForm-Feld, in das Sie Werte eingeben können, wenn die Bedingungen =, !=, <, >, <= oder >= ausgewählt werden. Falls in der Spalte „Bedingung“ entweder „=META-Tag“ oder „!=META-Tag“ ausgewählt wurde, enthält die Spalte „Wert“ eine Dropdown-Liste mit den zur Auswahl stehenden META-Tags. Sie können alle Zeichen eingeben, mit folgenden Ausnahmen:
 - Es werden nie einfache Anführungszeichen eingegeben.
 - Als Platzhalter können Stern (*) und Punkt (.) verwendet werden. Bei Verwendung regulärer Ausdrücke können diese Zeichen an einer beliebigen Position in der Zeichenkette vorkommen.
 - Es werden keine Escape-Zeichen verwendet, d. h. Platzhalter werden nicht mit Escape-Zeichen versehen.
 - und/oder – Sie können durch Klicken zwischen den einzelnen Feldern wechseln. Werden mehrere Bedingungen in der Tabelle angegeben, können Sie mithilfe der Schaltflächen „und“ und „oder“ angeben, ob alle Bedingungen oder nur eine einzelne Bedingung erfüllt werden müssen. Mit „und“ geben Sie an, dass alle Bedingungen erfüllt werden müssen. Mit „oder“ geben Sie an, dass nur eine einzelne Bedingung erfüllt werden muss.

HINWEIS: Ihre Auswahl wird nur gültig, wenn in der Tabelle wenigstens zwei Zeilen enthalten sind. Alle Zeilen mit Ausnahme der letzten Zeile in der Tabelle weisen standardmäßig diesen logischen Operator auf. Die Kombination von „und“ bzw. „oder“ zwischen Zeilen in der Tabelle ist nicht möglich.

- Schaltflächen „+/-“ Mit der Schaltfläche „+“ wird am Ende der Tabelle eine zusätzliche Zeile hinzugefügt. Mit der Schaltfläche „-“ wird die ausgewählte Zeile aus der Tabelle entfernt. Die Position dieser Zeile innerhalb der Tabelle ist dabei ohne Bedeutung.
- Welche Ereignisse sollen im Musterabgleich ausgeschlossen werden – Geben Sie mithilfe dieser Tabelle die Bedingungen an, um festzulegen, welche Ereignisse die Korrelationsregel nicht auslösen.
 - Ereigniskennung – Eine Liste mit verfügbaren Ereigniskennungen, für die eine Korrelation erfolgen kann.
 - Bedingung – Die Spalte „Bedingung“ ist eine Dropdown-Liste mit Operatoren, die zum Erstellen einer Korrelationsbedingung verwendet werden können.
 - Wert – Bei der Spalte „Wert“ handelt es sich um ein FreeForm-Feld, in das Sie Werte eingeben können, wenn die Bedingungen =, !=, <, >, <= oder >= ausgewählt werden. Falls in der Spalte „Bedingung“ entweder „=META-Tag“ oder „!=META-Tag“ ausgewählt wurde, enthält die Spalte „Wert“ eine Dropdown-Liste mit den zur Auswahl stehenden META-Tags. Sie können alle Zeichen eingeben, mit folgenden Ausnahmen:
 - Es werden nie einfache Anführungszeichen eingegeben.
 - Als Platzhalter können Stern (*) und Punkt (.) verwendet werden. Bei Verwendung regulärer Ausdrücke können diese Zeichen an einer beliebigen Position in der Zeichenkette vorkommen.

- Es werden keine Escape-Zeichen verwendet, d. h. Platzhalter werden nicht mit Escape-Zeichen versehen.
- und/oder – Sie können durch Klicken zwischen den einzelnen Feldern wechseln. Werden mehrere Bedingungen in der Tabelle angegeben, können Sie mithilfe der Schaltflächen „und“ und „oder“ angeben, ob alle Bedingungen oder nur eine einzelne Bedingung erfüllt werden müssen. Mit „und“ geben Sie an, dass alle Bedingungen erfüllt werden müssen. Mit „oder“ geben Sie an, dass nur eine einzelne Bedingung erfüllt werden muss.

HINWEIS: Ihre Auswahl wird nur gültig, wenn in der Tabelle wenigstens zwei Zeilen enthalten sind. Alle Zeilen mit Ausnahme der letzten Zeile in der Tabelle weisen standardmäßig diesen logischen Operator auf. Die Kombination von „und“ bzw. „oder“ zwischen Zeilen in der Tabelle ist nicht möglich.

- Schaltflächen „+/-“ – Mit der Schaltfläche „+“ wird am Ende der Tabelle eine zusätzliche Zeile hinzugefügt. Mit der Schaltfläche „-“ wird die ausgewählte Zeile aus der Tabelle entfernt. Die Position dieser Zeile innerhalb der Tabelle ist dabei ohne Bedeutung.

Definition von Korrelationsregeln

Assistenten für Korrelationsregeln: Mithilfe von [Beobachtungsliste](#), [Grundlegende Korrelation](#) und [Erweiterte Korrelation](#) können Sie schnell vordefinierte Regeltypen hinzufügen, je nachdem, welche Korrelation Sie ausführen möchten. Das Erzeugen der Korrelationsregel in der nativen Regelsprache von Correlation Engine erfolgt über den Assistenten der einzelnen Regeltypen. Jede dieser Regeln wird mithilfe des Fensters „Korrelationsregeln“ unter der Registerkarte „Admin“ erstellt.

Der Regelassistent umfasst auch einen FreeForm-Editor, mit dem Sie die [RuleLg](#)-Sprache für die Korrelationsdefinition verwenden können, um eine Regel direkt der nativen Regelsprache von Correlation Engine hinzuzufügen.

Beobachtungsliste

Es kann aus vier unterschiedlichen Filtertypen ausgewählt werden. Dazu gehören:

- Alle zulassen – Äquivalent zum Ausführen eines Filtergrads größer oder gleich null (0).
- Muster – Alle regulären Ausdrücke mit einer Grep-Syntax.
- Filtermanager – Eine Dropdown-Liste, die den Filtermanager anzeigt, um einen neuen Filter auszuwählen oder zu erstellen.
- Editor – Erstellen von Kriterien zum Einschließen oder Ausschließen von Ereignissen auf Grundlage boolescher Algebra.

Weitere Informationen erhalten Sie unter [Erstellen einer Regel für die Beobachtungsliste](#).

Grundlegende Korrelation

Es kann aus vier unterschiedlichen Filtertypen ausgewählt werden. Dazu gehören:

- Alle zulassen – Äquivalent zum Ausführen eines Filtergrads größer oder gleich null (0).
- Muster – Alle regulären Ausdrücke mit einer Grep-Syntax.
- Filtermanager – Eine Dropdown-Liste, die den Filtermanager anzeigt, um einen neuen Filter auszuwählen oder zu erstellen.
- Editor – Erstellen von Kriterien zum Einschließen oder Ausschließen von Ereignissen auf Grundlage boolescher Algebra.

Mithilfe dieser Regel können Sie zählen, wie häufig bestimmte Bedingungen innerhalb eines bestimmten Zeitrahmens erfüllt werden.

Beispielsweise kann mit einer grundlegenden Korrelationsregel nach derselben Quellen-IP-Adresse gesucht werden, die innerhalb von 5 Minuten fünfmal gemeldet wurde, auch wenn die Ereignisse über verschiedene Produkte (z. B. IDS (Intrusion Detection System) und Firewall) gemeldet werden.

Weitere Informationen erhalten Sie unter [Erstellen einer grundlegenden Korrelationsregel](#).

Erweiterte Korrelation

Es kann aus vier unterschiedlichen Filtertypen ausgewählt werden. Dazu gehören:

- Alle zulassen - Äquivalent zum Ausführen eines Filtergrads größer oder gleich null (0).
- Muster – Alle regulären Ausdrücke mit einer Grep-Syntax.
- Filtermanager – Eine Dropdown-Liste, die den Filtermanager anzeigt, um einen neuen Filter auszuwählen oder zu erstellen.
- Editor – Erstellen von Kriterien zum Einschließen oder Ausschließen von Ereignissen auf Grundlage boolescher Algebra.

Mit dieser Regel können Sie Folgendes ausführen:

- Zählen, wie häufig bestimmte Bedingungen innerhalb eines bestimmten Zeitrahmens erfüllt werden.
- Alle Funktionen einer einfachen Korrelationsregel einbeziehen und Ereignisse mit früheren Ereignissen abgleichen.

Beispielsweise kann mit einer erweiterten Korrelationsregel nach Ereignissen gesucht werden, die von derselben Quellen-IP-Adresse zu derselben Ziel-IP-Adresse erfolgten, denselben Ereignisnamen aufweisen und innerhalb und außerhalb der Firewall aufgetreten sind. (Dies bedeutet, dass möglicherweise ein Angriff durch die Firewall gelangen konnte.)

Weitere Informationen erhalten Sie unter [Erstellen einer erweiterten Korrelationsregel](#).

FreeForm RuleLg-Korrelation

Mit der RuleLg-Sprache für die Definition der Korrelationsregel können Sie die Definition von Korrelationsregeln selbstständig steuern. Sie sollten mit der RuleLg-Sprache für die Definition der Korrelationsregel vertraut sein, bevor Sie diesen Korrelationsregeltyp verwenden.

Weitere Informationen erhalten Sie unter [Erstellen einer FreeForm RuleLg-Korrelationsregel](#).

Erstellen einer Regel für die Beobachtungsliste

Erstellen Sie eine Regel für die Beobachtungsliste, wenn Sie eine Zeichenkette angeben möchten, die von der Correlation Engine auf eingehende Ereignisse überprüft werden soll. So erstellen Sie eine Regel für die Beobachtungsliste:

- Wählen Sie im ersten Fenster „Assistent für Korrelationsregeln“ die Regel für die Beobachtungsliste aus. Geben Sie folgende Informationen an:
 - Regelname – In der Liste mit Regeln angezeigter Name. Max. 255 Zeichen zulässig. Punkte sind nicht zulässig. Erweiterte ASCII-Zeichen sind nicht zulässig. Für den Regelnamen muss Groß-/Kleinschreibung beachtet werden.
 - Beschreibung – Kurze Beschreibung. Max. 1.024 Zeichen für den beschreibenden Text zulässig.

- Filtertyp
 - Alle zulassen -
 - Muster – Ereignis beobachten mit *

Filtertyp: **Pattern**

Filterdefinition

Ereignisse suchen, die enthalten*:

** Verwenden Sie für den Musterabgleich reguläre Ausdrücke.

- Filtermanager – ({Eigentümer-ID}:{Filtername}):<Feldname>

Filtertyp: **Filter Manager**

Filterdefinition

Ausgewählter Filter*:

** Erstellen Sie einen Filter oder wählen Sie einen Filter aus dem Filtermanager aus.

- Editor

Filtertyp: **Builder**

Filterdefinition

Diese Ereignisse sollten eingeschlossen w... ☒ Und ☐ Oder

Ereigniskennung	Bedingung	Wert	und/oder

+
-

Diese Ereignisse sollten ausgeschlossen ... ☒ Und ☐ Oder

Ereigniskennung	Bedingung	Wert	und/oder

+
-

- Seite „Korreliertes Ereignis und korrelierte Aktionen“ – In diesem Feld wird angegeben, welche Aktion automatisch ausgeführt wird, wenn Ereignisse mit dieser

Korrelationsregel übereinstimmen. Es muss nur der Schweregrad eingegeben werden. Dieser ist standardmäßig auf Schweregrad 4 festgelegt.

- Ereignisname – Standard: Korreliertes Ereignis. Damit wird der Name des korrelierten Ereignisses angegeben.
- Ressource – Standard: Correlation Engine. Damit wird der Name einer Ressource im System angegeben.
- Teilressource – Standard: <Keine>. Damit wird bei Ressourcen mit mehreren Teilressourcen der Name der Teilressource angegeben.
- Schweregrad einstellen auf: – Standard: 4. Dies ist der Schweregrad, der diesem Ereignis zugewiesen wird. Gültige Werte sind 0, 1, 2, 3, 4 (Standard) und 5. Es wird eine Dropdown-Liste mit den zulässigen Schweregraden angegeben.
- Benutzerdefinierter Nachrichtentext – Standard: <Keine>. Dieser Text wird zusammen mit dem Ereignis angezeigt. Er ist hilfreich, um die Bedingung zu identifizieren, die die Regel für die Beobachtungsliste ausgelöst hat. Max. 4.000 Zeichen zulässig. Der in diesem Feld eingegebene Text wird dem Text des korrelierten Ereignisses mit einer Pipe-Trennung vorangestellt. Beispielsweise würde für die Eingabe „Neue Nachricht“ die korrelierte Nachricht „Neue Nachricht|Drei Instanzen von...“ erstellt werden.
- Aktion durchführen (nur Oracle) – Standard: <Keine>. Dies ist der Name einer ausführbaren Datei, die beim Auslösen der Regel für die Beobachtungsliste ausgeführt wird. Diese Datei muss sich im Verzeichnis \$ESEC_HOME/sentinel/exec befinden und vom Benutzer „esecadm“ ausgeführt werden können. In diesem FreeForm-Textfeld erfolgt keine Eingabeüberprüfung. Sie können die META-Tags angeben, die an die ausführbare Datei gesendet werden sollen.
- Aktion durchführen (nur MSSQL) – Standard: <Keine>. Dies ist der Name einer ausführbaren Datei, die beim Auslösen der Regel ausgeführt wird. Diese Datei muss sich im Verzeichnis %ESEC_HOME%\sentinel\bin befinden und vom Benutzer „esecadm“ ausgeführt werden können. Es erfolgt keine Eingabeüberprüfung. Sie können die META-Tags angeben, die an die ausführbare Datei gesendet werden sollen. Nachfolgend sind zwei Beispiele angegeben: für eine Korrelationsregel zum Senden einer Email und für eine Korrelationsregel zum Senden des korrelierten Ereignisses an HP OVO.

Neue Korrelationsregel [S] [W] [R] [B] [C] [D] [X]

Korreliertes Ereignis und korrelierte Aktionen

Konfigurieren Sie das korrelierte Ereignis und die Aktionen, die diese Regel auslösen.

Korreliertes Ereignis

Ereignisname:

Ressource:

Teilressource:

Schweregrad:

Nachricht:

Aktionen

Aktion durchführen:

Vorfall erstellen ☐ iTRAC-Prozess beifüg...

Die Befehlszeile und die Parameterzeile werden als Zeichenkette eingelesen. Beim Parsing wird „,\" (umgekehrter Schrägstrich) als Escape-Zeichen behandelt. Dieser umgekehrte Schrägstrich kann als Escape-Zeichen für folgende Zeichen verwendet werden: \, % und “. Beispielsweise ist \%\" äquivalent zu %\". Falls Sie einen Befehl mit umgekehrtem Schrägstrich verwenden, z. B. zum Ausführen eines Windows-Befehls in einem Unterverzeichnis von „sentinel\bin“, ist es erforderlich, zwei umgekehrte Schrägstriche (\\) für jeden Verzeichnisschrägstrich einzugeben. Beispielsweise muss zum Ausführen der Stapeldatei „run.bat“, die sich im Verzeichnis %esec_home%\sentinel\bin\batchfiles\ befindet, Folgendes eingegeben werden: batchfiles\\run.bat. Beachten Sie, dass sich alle ausführbaren Dateien unter %esec_home%\sentinel\bin\ befinden müssen.

Konfiguration der Korrelationsaktion

Aktionsname:

Beschreibung:

Befehl:

Parameter:

Konfiguration der Korrelationsaktion

Aktionsname:

Beschreibung:

Befehl:

Parameter:

OK Abbrechen Hilfe

HINWEIS: Weitere Informationen zu Befehlen und Parametern finden Sie im Referenzhandbuch für Benutzer in "Kapitel 5 – META-Tags für Wizard und Sentinel" sowie im Abschnitt [Korrelationsausgabe](#).

- Vorfall erstellen – Eine Aktion des korrelierten Ereignisses kann auch das Erstellen eines Vorfalls sein.
- iTrac-Prozess beifügen – Dem erstellten Vorfall kann ein iTrac-Prozess beigelegt sein.

Erstellen einer grundlegenden Korrelationsregel

Erstellen Sie eine grundlegende Korrelationsregel, wenn Sie zählen möchten, wie häufig bestimmte Bedingungen innerhalb eines bestimmten Zeitrahmens erfüllt werden. Es werden dazu die folgenden Schritte ausgeführt:

- Wählen Sie im ersten Fenster „Assistent für Korrelationsregeln“ die grundlegende Korrelationsregel aus. Geben Sie folgende Informationen an:
 - Regelname – In der Liste mit Regeln angezeigter Name. Max. 255 Zeichen zulässig. Punkte sind nicht zulässig. Erweiterte ASCII-Zeichen sind nicht zulässig. Für den Regelnamen muss Groß-/Kleinschreibung beachtet werden.
 - Beschreibung – Kurze Beschreibung. Max. 1.024 Zeichen für den beschreibenden Text zulässig.
- Filtertyp
 - Alle zulassen
 - Muster

Filtertyp

Filterdefinition

Ereignisse suchen, die enthalten*:

** Verwenden Sie für den Musterabgleich reguläre Ausdrücke.

- Filtermanager – ({Eigentümer-ID}:{Filtername}):<Feldname>

Filtertyp: Pattern

Filterdefinition

Ereignisse suchen, die enthalten*:

** Verwenden Sie für den Musterabgleich reguläre Ausdrücke.

▫ Editor

Filtertyp: Builder

Filterdefinition

Diese Ereignisse sollten eingeschlossen w... ☒ Und ☐ Oder

Ereigniskennung	Bedingung	Wert	und/oder

+
-

Diese Ereignisse sollten ausgeschlossen ... ☒ Und ☐ Oder

Ereigniskennung	Bedingung	Wert	und/oder

+
-

- Schwellenwert und Gruppierungskriterien (obere Fensterhälfte) – Regel aktivieren: – Mit dieser Option können Sie „Abgleich“-Kriterien für mehrere Ereignisse eingeben, die für einen bestimmten Zeitraum im System eingehen.
 - Wenn Bedingung _ Mal erfüllt ist - Standard: 1. Eine Regel wird nur ausgelöst, wenn sie mit der angegebenen Häufigkeit ermittelt wurde. Der gültige Eingabebereich für diesen Schwellenwert ist 1 oder größer.
 - innerhalb (Zeitraumen) – Standard: 60 Sekunden. Damit wird die Bedingung an den Zeitrahmen gebunden. Dies ist eine kombinierte, variable Eingabe und eine Dropdown-Liste. Die Optionen für die Dropdown-Liste sind: Sekunden, Minuten, Stunden und Tage.

HINWEIS: Wenn der Zeitrahmen gleich null (0) ist, erfolgt der Auslöser unmittelbar. Bei einer grundlegenden Korrelation tritt das Ereignis max. ein Mal im sofortigen Zeitrahmen auf.

- Schwellenwert und Gruppierungskriterien (untere Fensterhälfte) – Korrelation erfolgt bei verschiedenen Kombinationen der folgenden META-Tags – Wählen Sie die META-Tags aus, die in Kombination für die Korrelation verwendet werden sollen. Ereignisse werden anhand der ausgewählten META-Tags in Gruppen unterteilt.

- Seite „Korreliertes Ereignis und korrelierte Aktionen“ – In diesem Feld wird angegeben, welche Aktion automatisch ausgeführt wird, wenn Ereignisse mit dieser Korrelationsregel übereinstimmen. Es muss nur der Schweregrad eingegeben werden. Dieser ist standardmäßig auf Schweregrad 4 festgelegt.
 - Ereignisname – Standard: Korreliertes Ereignis. Damit wird der Name des korrelierten Ereignisses angegeben.
 - Ressource – Standard: Correlation Engine. Damit wird der Name einer Ressource im System angegeben.
 - Teilressource – Standard: <Keine>. Damit wird bei Ressourcen mit mehreren Teilressourcen der Name der Teilressource angegeben.
 - Schweregrad einstellen auf: – Standard: 4. Dies ist der Schweregrad, der diesem Ereignis zugewiesen wird. Gültige Werte sind 0, 1, 2, 3, 4 (Standard) und 5. Es wird eine Dropdown-Liste mit den zulässigen Schweregraden angegeben.
 - Benutzerdefinierter Nachrichtentext – Standard: <Keine>. Dieser Text wird zusammen mit dem Ereignis angezeigt. Er ist hilfreich, um die Bedingung zu identifizieren, die die Regel für die Beobachtungsliste ausgelöst hat. Max. 4.000 Zeichen zulässig. Der in diesem Feld eingegebene Text wird dem Text des korrelierten Ereignisses mit einer Pipe-Trennung vorangestellt. Beispielsweise würde für die Eingabe „Neue Nachricht“ die korrelierte Nachricht „Neue Nachricht|Drei Instanzen von...“ erstellt werden.

- Diesen Befehl ausführen (nur Oracle) – Standard: <Keine>. Dies ist der Name einer ausführbaren Datei, die beim Auslösen der Regel für die Beobachtungsliste ausgeführt wird. Diese Datei muss sich im Verzeichnis \$ESEC_HOME/sentinel/exec befinden und vom Benutzer „esecadm“ ausgeführt werden können. In diesem FreeForm-Textfeld erfolgt keine Eingabeüberprüfung. Sie können die META-Tags angeben, die an die ausführbare Datei gesendet werden sollen.
- Aktion durchführen (nur MSSQL) – Standard: <Keine>. Dies ist der Name einer ausführbaren Datei, die beim Auslösen der Regel ausgeführt wird. Diese Datei muss sich im Verzeichnis %ESEC_HOME%\sentinel\bin befinden und vom Benutzer „esecadm“ ausgeführt werden können. Es erfolgt keine Eingabeüberprüfung. Sie können die META-Tags angeben, die an die ausführbare Datei gesendet werden sollen. Nachfolgend sind zwei Beispiele angegeben: für eine Korrelationsregel zum Senden einer Email und für eine Korrelationsregel zum Senden des korrelierten Ereignisses an HP OVO.

Neue Korrelationsregel S W R B C D X

Korreliertes Ereignis und korrelierte Aktionen

Konfigurieren Sie das korrelierte Ereignis und die Aktionen, die diese Regel auslösen.

Korreliertes Ereignis

Ereignisname: Correlated Event

Ressource: Correlation Engine

Teilressource:

Schweregrad: 4 - Erheblich

Nachricht:

Aktionen

Aktion durchführen: Konfigurieren...

Vorfall erstellen ☐ iTRAC-Prozess beifüg... NONE

Zurück Fertig stellen Abbrechen

Konfiguration der Korrelationsaktion

Aktionsname:

Beschreibung:

Befehl:

Parameter:

OK Abbrechen Hilfe

Konfiguration der Korrelationsaktion

Aktionsname:

Beschreibung:

Befehl:

Parameter:

OK Abbrechen Hilfe

HINWEIS: Weitere Informationen zu Befehlen und Parametern finden Sie im Referenzhandbuch für Benutzer in „Kapitel 5 – META-Tags für Wizard und Sentinel“ sowie im Abschnitt [Korrelationsausgabe](#).

- Vorfall erstellen – Eine Aktion des korrelierten Ereignisses kann auch das Erstellen eines Vorfalls sein.
- iTrac-Prozess beifügen – Dem erstellten Vorfall kann ein iTrac-Prozess beigelegt sein.

Erstellen einer erweiterten Korrelationsregel

Mit einer erweiterten Korrelationsregel können Sie eine Regel komplexer gestalten, indem Sie im Fenster „Weitere Kriterien“ eine zusätzliche Bedingung einfügen und der Regeldefinition damit eine Ebene logischer UND-Verknüpfungen hinzufügen.

Erstellen Sie eine erweiterte Korrelationsregel, wenn Sie nicht nur zählen möchten, wie häufig bestimmte Bedingungen erfüllt werden, sondern auch eine Meldung erhalten möchten, wenn Ereignisse (auch frühere Ereignisse) die angegebenen Kriterien erfüllen. Es werden dazu die folgenden Schritte ausgeführt:

- Wählen Sie im ersten Fenster „Assistent für Korrelationsregeln“ die erweiterte Korrelationsregel aus. Geben Sie folgende Informationen an:
 - Regelname – In der Liste mit Regeln angezeigter Name. Max. 255 Zeichen zulässig. Punkte sind nicht zulässig. Erweiterte ASCII-Zeichen sind nicht zulässig. Für den Regelnamen muss Groß-/Kleinschreibung beachtet werden.

- Beschreibung – Kurze Beschreibung. Max. 1.024 Zeichen für den beschreibenden Text zulässig.
- Filtertyp
 - Alle zulassen
 - Muster

Filtertyp: Pattern

Filterdefinition

Ereignisse suchen, die enthalten*:

** Verwenden Sie für den Musterabgleich reguläre Ausdrücke.

- Filtermanager – ({Eigentümer-ID}:{Filtername}):<Feldname>

Filtertyp: Filter Manager

Filterdefinition

Ausgewählter Filter*:

** Erstellen Sie einen Filter oder wählen Sie einen Filter aus dem Filtermanager aus.

- Editor

Filtertyp: Builder

Filterdefinition

Diese Ereignisse sollten eingeschlossen w... ☒ Und ☐ Oder

Ereigniskennung	Bedingung	Wert	und/oder

+
-

Diese Ereignisse sollten ausgeschlossen ... ☒ Und ☐ Oder

Ereigniskennung	Bedingung	Wert	und/oder

+
-

Zurück Weiter Abbrechen

- Weitere Kriterien – Mit dieser Option können Sie „Abgleich“-Kriterien für mehrere Ereignisse eingeben, die für einen bestimmten Zeitraum im System eingehen. Die Standardzeit beträgt 60 Sekunden. Dies ist eine kombinierte, variable Eingabe und eine Dropdown-Liste. Die Optionen für die Dropdown-Liste sind: Sekunden, Minuten, Stunden und Tage.
- Schwellenwert und Gruppierungskriterien (obere Fensterhälfte) – Regel aktivieren: – Mit dieser Option können Sie „Abgleich“-Kriterien für mehrere Ereignisse eingeben, die für einen bestimmten Zeitraum im System eingehen.
 - Wenn Bedingung _ Mal erfüllt ist - Standard: 1. Eine Regel wird nur ausgelöst, wenn sie mit der angegebenen Häufigkeit ermittelt wurde. Der gültige Eingabebereich für diesen Schwellenwert ist 1 oder größer.
 - innerhalb (Zeitraumen) – Standard: 60 Sekunden. Damit wird die Bedingung an den Zeitraumen gebunden. Dies ist eine kombinierte, variable Eingabe und eine Dropdown-Liste. Die Optionen für die Dropdown-Liste sind: Sekunden, Minuten, Stunden und Tage.

HINWEIS: Wenn der Zeitraumen gleich null (0) ist, erfolgt der Auslöser unmittelbar. Bei einer grundlegenden Korrelation tritt das Ereignis max. ein Mal im sofortigen Zeitraumen auf.

- Schwellenwert und Gruppierungskriterien (untere Fensterhälfte) – Korrelation erfolgt bei verschiedenen Kombinationen der folgenden META-Tags – Wählen Sie die META-Tags aus, die in Kombination für die Korrelation verwendet werden sollen. Ereignisse werden anhand der ausgewählten META-Tags in Gruppen unterteilt.

Neue Korrelationsregel [S] [W] [R] [B] [C] [D] [X]

Schwellenwert und Gruppierungskriterien

"Wählen Sie den Schwellenwert, die Dauer und die Gruppierungskriterien aus."

Schwellenwert

Auslösen, wenn Bedingung erfüllt ist Uhrzeit(en) innerhalb

pro Gruppe

Ähnliche Ereignisse nach folgenden Meta-Tags gruppieren

AttackId
ControlMonitor
ControlPack
CorrelatedEventUids
Criticality
Ct1
Ct2
Ct3
CustomerVar1

Hinzufügen >>

<< Entfernen

Zurück Weiter Abbrechen

- Seite „Korreliertes Ereignis und korrelierte Aktionen“ – In diesem Feld wird angegeben, welche Aktion automatisch ausgeführt wird, wenn Ereignisse mit dieser Korrelationsregel übereinstimmen. Es muss nur der Schweregrad eingegeben werden. Dieser ist standardmäßig auf Schweregrad 4 festgelegt.
 - Ereignisname – Standard: Korreliertes Ereignis. Damit wird der Name des korrelierten Ereignisses angegeben.
 - Ressource – Standard: Correlation Engine. Damit wird der Name einer Ressource im System angegeben.
 - Teilressource – Standard: <Keine>. Damit wird bei Ressourcen mit mehreren Teilressourcen der Name der Teilressource angegeben.
 - Schweregrad einstellen auf: – Standard: 4. Dies ist der Schweregrad, der diesem Ereignis zugewiesen wird. Gültige Werte sind 0, 1, 2, 3, 4 (Standard) und 5. Es wird eine Dropdown-Liste mit den zulässigen Schweregraden angegeben.
 - Benutzerdefinierter Nachrichtentext – Standard: <Keine>. Dieser Text wird zusammen mit dem Ereignis angezeigt. Er ist hilfreich, um die Bedingung zu identifizieren, die die Regel für die Beobachtungsliste ausgelöst hat. Max. 4.000 Zeichen zulässig. Der in diesem Feld eingegebene Text wird dem Text des korrelierten Ereignisses mit einer Pipe-Trennung vorangestellt. Beispielsweise würde für die Eingabe „Neue Nachricht“ die korrelierte Nachricht „Neue Nachricht|Drei Instanzen von...“ erstellt werden.
 - Diesen Befehl ausführen (nur Oracle) – Standard: <Keine>. Dies ist der Name einer ausführbaren Datei, die beim Auslösen der Regel für die Beobachtungsliste ausgeführt wird. Diese Datei muss sich im Verzeichnis \$ESEC_HOME/sentinel/exec befinden und vom Benutzer „esecadm“ ausgeführt werden können. In diesem FreeForm-Textfeld erfolgt keine Eingabeüberprüfung. Sie können die META-Tags angeben, die an die ausführbare Datei gesendet werden sollen.
 - Aktion durchführen (nur MSSQL) – Standard: <Keine>. Dies ist der Name einer ausführbaren Datei, die beim Auslösen der Regel ausgeführt wird. Diese Datei muss sich im Verzeichnis %ESEC_HOME%\sentinel\bin befinden und vom Benutzer „esecadm“ ausgeführt werden können. Es erfolgt keine Eingabeüberprüfung. Sie können die META-Tags angeben, die an die ausführbare Datei gesendet werden sollen. Nachfolgend sind zwei Beispiele angegeben: für eine Korrelationsregel zum Senden einer Email und für eine Korrelationsregel zum Senden des korrelierten Ereignisses an HP OVO.

Neue Korrelationsregel

S | W | R | B | C | D | X

Korreliertes Ereignis und korrelierte Aktionen

Konfigurieren Sie das korrelierte Ereignis und die Aktionen, die diese Regel auslösen.

Korreliertes Ereignis

Ereignisname: Correlated Event

Ressource: Correlation Engine

Teilressource:

Schweregrad: 4 - Erheblich

Nachricht:

Aktionen

Aktion durchführen: Konfigurieren...

Vorfall erstellen

☐ iTRAC-Prozess beifüg...

NONE

Zurück

Fertig stellen

Abbrechen

Konfiguration der Korrelationsaktion

Aktionsname: email_me

Beschreibung: email_me.

Befehl: email_interface.csh

Parameter: %all% <name>@<domain.name> "telnet hit"

OK

Abbrechen

Hilfe

Konfiguration der Korrelationsaktion

Aktionsname:	send to HP OVO
Beschreibung:	send to HP OVO
Befehl:	esec_ovo
Parameter:	%all%

OK Abbrechen Hilfe

HINWEIS: Weitere Informationen zu Befehlen und Parametern finden Sie im Referenzhandbuch für Benutzer in „Kapitel 5 – META-Tags für Wizard und Sentinel“ sowie im Abschnitt [Korrelationsausgabe](#).

- Vorfall erstellen – Eine Aktion des korrelierten Ereignisses kann auch das Erstellen eines Vorfalls sein.
- iTrac-Prozess beifügen – Dem erstellten Vorfall kann ein iTrac-Prozess beigefügt sein.

Erstellen einer FreeForm RuleLg-Korrelationsregel

Die Correlation Engine umfasst drei grundlegende Operationen. Diese Operationen werden kombiniert, sodass eine Regel mit Operatoren für Fluss, Vereinigung und Schnittmenge gebildet wird: Die drei grundlegenden Operationen lauten:

- [Filteroperation](#)
- [Fensteroperation](#)
- [Auslöseoperation](#)

ACHTUNG: Falls Sie ein Tag umbenannt haben, sollte zum Erstellen einer Korrelationsregel nicht der ursprüngliche Name verwendet werden.

In der Regelsprache werden diese Operationen und die intuitiven Kombinationsmöglichkeiten zum Definieren von Korrelationsregeln direkt widerspiegelt. Jede einzelne Operation wurde speziell konzipiert und implementiert, um eine hohe Leistung zu erzielen, und funktioniert für einen bestimmten Satz von Ereignissen: einen Ereignissatz als Eingabe empfangen und einen Ereignissatz ausgeben. Das aktuell von einer Regel verarbeitete Ereignis hat häufig eine spezielle Bedeutung für die Semantik der Sprache. Das aktuelle Ereignis ist immer Teil des Ereignissatzes als Ein- oder Ausgabe einer Operation, es sei denn, der Satz ist leer. Wenn der Eingabesatz einer Operation leer ist, wird die Operation nicht evaluiert.

Vereinfacht gesagt, verarbeitet eine Korrelationsregel die in die Correlation Engine-Instanz eingehenden Ereignisse nacheinander in der entsprechenden Reihenfolge. Tatsächlich kann die Correlation Engine-Instanz mehrere Ereignisse verarbeiten und zugleich mehrere Regeln für ein Ereignis evaluieren.

New Correlation Rule

Free Form RuleLg

Enter your correlation rule using the RuleLg language.

Correlation Rule ?

Validation Output

< Back Next > Cancel

Filteroperation

Mithilfe von Filteroperationen (boolescher Ausdruck) ist eine Filterung gemäß dem Inhalt des aktuellen Ereignisses möglich, d. h. anhand seiner META-Tag-Werte und des vom Filter angegebenen booleschen Ausdrucks. Bei der Ausgabe eines Filters handelt es sich entweder um einen leeren Satz (wenn das aktuelle Ereignis nicht mit dem Filter übereinstimmt) oder um einen Satz, der das aktuelle Ereignis und alle anderen Ereignisse aus dem eingehenden Satz enthält.

- Filter arbeiten mit dem aktuellen Ereignis, indem sie den booleschen Ausdruck für das aktuelle Ereignis evaluieren:
 - Die Filteroperation gibt den Eingabesatz zurück, wenn der boolesche Ausdruck als „true“ (wahr) ausgewertet wird.
 - Die Filteroperation gibt den leeren Satz zurück, wenn der boolesche Ausdruck als „false“ (falsch) ausgewertet wird.
- Der boolesche Ausdruck ist eine Zusammenstellung von Vergleichs- und Abgleichsanweisungen mit folgenden booleschen Operatoren: „and“ (und), „or“ (oder) und „not“ (nicht).

Filteroperation – Vorrang und Assoziativität von RuleLg-Operatoren

Vorrang der booleschen Filteroperatoren (in der Reihenfolge vom höchsten zum niedrigsten Vorrang):

Operator	Bedeutung	Operatortyp	Assoziativität
not	Logisches Nicht	Unär	Keine
and	Logisches Und	Binär	Von links nach rechts
or	Logisches Oder	Binär	Von links nach rechts

Es gilt Folgendes:

- Die Vergleichsanweisungen ermöglichen die Evaluierung von Ereignis-META-Tag-Werten mit anderen Ereignis-META-Tag-Werten bzw. -Ausnahmen.
- Folgende Vergleichsoperatoren stehen zur Verfügung: =, !=, >, <, >=, <=.
- Bei den verfügbaren Abgleichsanweisungen handelt es sich um „reguläre Ausdrücke abgleichen“, `match regex()`, bzw. um „Teilnetze abgleichen“, `match subnet()`.
- Vergleichs- und Abgleichsanweisungen können mithilfe von Klammern beliebig tief verschachtelt werden.
- Den META-Tag-Namen in Vergleichs- und Abgleichsanweisungen muss stets "e." vorangestellt werden, um das aktuelle Ereignis anzugeben.
- Wenn es sich bei einem Filter um die letzte (oder einzige) Operation einer Korrelationsregel handelt, wird der Ausgabesatz des Filters verwendet, um ein korreliertes Ereignis zu erstellen (die korrelierten Ereignisse sind der Ausgabesatz an Ereignissen der Filteroperation, wobei das aktuelle Ereignis an erster Stelle steht).
- Wenn es sich bei einem Filter nicht um die letzte Operation einer Korrelationsregel handelt (d. h., wenn sich rechts davon ein Flussoperator befindet), wird der Ausgabesatz des Filters als Eingabesatz für andere Operationen (über den Flussoperator) verwendet.

Beispiel: Wenn das aktuelle Ereignis den Schweregrad 4 aufweist und das Ressourcen-META-Tag entweder „FW“ oder „Comm“ enthält, wird ein korreliertes Ereignis gesendet, wobei das aktuelle (einzige) Ereignis als korreliertes Ereignis aufgeführt wird.

```
filter(e.sev = 4 and (e.res match regex("FW") or e.res  
match regex("Comm")))
```

Wenn beispielsweise eines der aktuellen Ereignis-META-Tags „ABC“ enthält, wird ein korreliertes Ereignis gesendet, wobei das aktuelle (einzige) Ereignis als korreliertes Ereignis aufgeführt wird.

```
filter(e.all match regex("ABC"))
```

Fensteroperation („window“)

Eine Fensteroperation (`simple boolean expression[, filter expression], int duration`) wird auf das aktuelle Ereignis in Bezug auf ein Fenster früherer Ereignisse angewendet. Frühere Ereignisse werden von der Fensteroperation selbst aufrechterhalten. Die Ausgabe eines Fensters ist entweder ein leerer Satz (wenn das aktuelle Ereignis nicht mit dem einfachen booleschen Ausdruck übereinstimmt) oder ein Satz, der das aktuelle Ereignis und alle früheren Ereignisse enthält, für die der einfache boolesche Ausdruck wahr ist.

Der einfache boolesche Ausdruck ist entweder eine einzelne Vergleichsanweisung oder eine einzelne Abgleichsanweisung, die auf den META-Tag-Wert eines früheren Ereignisses und einen aktuellen META-Tag-Wert oder eine Konstante angewendet wird. Für boolesche Ausdrücke:

- META-Tag-Namen muss "e." vorangestellt werden, um das aktuelle Ereignis anzugeben, oder "w.", um die früheren Ereignisse anzugeben.
- Folgende Vergleichsoperatoren stehen zur Verfügung: =, !=, >, <, >=, <=, in und not in.
- Bei den verfügbaren Abgleichsanweisungen handelt es sich um „reguläre Ausdrücke abgleichen“, match regex(), bzw. um „Teilnetze abgleichen“, match subnet().
- Die Zeichenkette „w.[meta-tag]“ muss im einfachen booleschen Ausdruck eines Fensters vorhanden sein.
- Wenn ein früheres Ereignis mit dem aktuellen Ereignis für den einfachen booleschen Ausdruck als „true“ (wahr) ausgewertet wird, besteht der Ausgabesatz aus dem eingehenden Ereignis sowie allen Übereinstimmungen im Fenster.
- Wenn keine Ereignisse im Fenster mit dem aktuellen Ereignis für den einfachen booleschen Ausdruck übereinstimmen, wird ein leerer Satz ausgegeben.

Frühere Ereignisse werden für die angegebene Dauer der Fensteroperation aufrechterhalten.

Mit dem optionalen Filterausdrucksparameter eines Fensters kann gesteuert werden, welche Ereignisse im Fenster vorhanden sind. Bei diesem Ausdruck kann es sich um jeden gültigen Filter handeln.

- Jedes an Correlation Engine eingehende Ereignis, das diesen Filter passiert, wird in das Fenster der früheren Ereignisse aufgenommen.
- Wenn kein Filterausdruck vorhanden ist, werden alle bei der Correlation Engine-Instanz eingehenden Ereignisse vom Fenster aufrechterhalten.
- Das aktuelle Ereignis wird erst dann in das Fenster aufgenommen, wenn alle Fensterevaluationen für das aktuelle Ereignis abgeschlossen sind.
- Nur die relevanten Teile der früheren Ereignisse werden tatsächlich vom Fenster aufrechterhalten (um die Arbeitsspeicherauslastung zu verringern).

Wenn es sich bei einem Fenster um die letzte (oder einzige) Operation einer Korrelationsregel handelt, wird der Ausgabesatz des Fensters verwendet, um ein korreliertes Ereignis zu erstellen (die korrelierten Ereignisse sind der Ausgabesatz an Ereignissen der Fensteroperation, wobei das aktuelle Ereignis an erster Stelle steht).

Beispiel 1

```
window(e.sip = w.sip, filter(e.sip match subnet
    (<xxx.xxx.x.x/yy>)), 60)
```

Wenn das aktuelle Ereignis im oben angeführten Beispiel eine Quellen-IP in der Adresse xxx.xxx.x.x/yy mit einer CIDR-Teilnetzmaske aufweist und mit mindestens einem Ereignis übereinstimmt, das innerhalb der letzten 60 Sekunden eintrat, wird ein korreliertes Ereignis zusammen mit dem aktuellen Ereignis und allen abgeglichenen früheren Ereignissen als korrelierte Ereignisse gesendet (aktuelles Ereignis zuerst).

Beispiel 2

```
window(e.sip = w.dip, 3600) intersection
window(e.dp = w.dp, 3600) intersection
window(e.evt = w.evt, 3600)
```


Das obige Beispiel stellt einen Domino-Regeltyp dar. Ein Angreifer nutzt ein anfälliges System aus und verwendet es als Angriffsplattform.

Beispiel 3

```
filter(e.sev > 3) flow (window(e.sip = w.sip, filter
    (e.sev >3), 5) intersection window(e.evt = w.evt,
    filter(e.sev >3), 5) intersection window(e.dip =
    w.dip, filter(e.sev >3), 5) intersection window(e.sn!
    = w.sn, filter(e.sev > 3),5)
```

Das obige Beispiel stellt einen Regeltyp „innerhalb/außerhalb“ dar. Eine Angriffssignatur wird auf zwei IDSs (Intrusion Detection System) angezeigt (eines innerhalb, das andere außerhalb einer Firewall) und der Angriff weist einen Schweregrad von mehr als 3 auf.

Auslöseoperation („trigger“)

Der Hauptzweck einer Auslöseoperation besteht darin, die Anzahl der Ereignisse in einer bestimmten Dauer zu ermitteln. Wenn die angegebene Anzahl innerhalb der angegebenen Dauer erreicht wird, wird ein Ereignissatz ausgegeben, der alle vom Auslöser aufrechterhaltenen Ereignisse enthält; wenn nicht, wird der leere Satz ausgegeben.

- Die Auslöseoperation erhält als Eingabe einen Ereignissatz, der als Teil des Ausgabe-Ereignissatzes zurückgegeben wird, wenn die angegebene Anzahl, Dauer und Diskriminatoren der früheren Eingabesätze und des aktuellen Eingabesatzes die von der Auslöseoperation angegebenen Kriterien erfüllen.
- Die Anzahl ist ein ganzzahliger Wert, der angibt, wie viele Ereignisse innerhalb des Dauerfensters eintreten müssen, damit ein nichtleerer Satz ausgegeben wird.
- Die Dauer ist ein ganzzahliger Wert in Sekunden, der angibt, wie lange die Ereignisse von der Auslöseoperation aufrechterhalten werden.
- Wenn der Wert für die Dauer gleich null ist, vergleicht eine Auslöseoperation einfach nur die Anzahl der Ereignisse im Eingabesatz mit dem Wert für die Anzahl und gibt das aktuelle Ereignis aus, wenn die tatsächliche Anzahl größer oder gleich dem für die Anzahl angegebenen Wert ist.
- Wenn ein neuer Eingabe-Ereignissatz erhalten wird, verwirft der Auslöser zunächst die veralteten Ereignisse (Ereignisse, die länger aufrechterhalten wurden als unter „Dauer“ angegeben) und fügt dann das aktuelle Ereignis ein. Wenn die Anzahl der resultierenden Ereignisse größer oder gleich der angegebenen Anzahl ist, gibt der Auslöser einen Satz mit allen Ereignissen aus.
- Wenn es sich bei einem Auslöser um die letzte (oder einzige) Operation einer Korrelationsregel handelt, wird der Ausgabesatz des Auslösers verwendet, um ein korreliertes Ereignis zu erstellen (die korrelierten Ereignisse sind der Ausgabesatz an Ereignissen der Auslöseoperation, wobei das aktuelle Ereignis an erster Stelle steht).
- Wenn es sich bei einem Auslöser nicht um die letzte Operation einer Korrelationsregel handelt (d. h., wenn sich rechts davon ein Flussoperator befindet), wird der Ausgabesatz des Auslösers als Eingabesatz für andere Operationen (über den Flussoperator) verwendet.
- Nachdem die Kriterien der Auslöseoperation erstmals erfüllt sind (und daher die Auslöseoperation einen Ereignissatz ausgibt), gilt: Wenn die Kriterien innerhalb der Verfallszeit erneut erfüllt werden und mindestens eines der zuvor ausgegebenen Ereignisse enthalten ist und es sich bei dem Auslöser um die letzte (oder einzige) Operation handelt, erstellt Correlation Engine kein neues korreliertes Ereignis, sondern erstellt stattdessen eine Aktualisierung für das vorherige korrelierte Ereignis.

- Der Diskriminator (META-Tag-Liste) ist eine kommagetrennte Liste mit META-Tags. Eine Auslöseoperation enthält verschiedene Anzahl-Werte für jede verschiedene Kombination der Diskriminator-META-Tags.

Wenn beispielsweise 5 Ereignisse mit derselben Quellen-IP innerhalb von 10 Sekunden eintreten, wird ein korreliertes Ereignis mit den 5 Ereignissen als korrelierte Ereignisse gesendet (aktuelles Ereignis zuerst).

```
trigger(5, 10, discriminator(e.sip))
```

Durch die Verwendung der FreeForm-Regeloption können Ausdrücke mit ungebundener Komplexität erstellt werden, die jedoch möglicherweise keinen Sinn ergeben. Die unterstützte normale Form eines RuleLg-Ausdrucks wird in drei Teile untergliedert: die Abschnitte „filter“, „window“ und „trigger“. Die drei Abschnitte werden durch einen Flussoperator verbunden.

Im Abschnitt „filter“ können mehrere verbundene Filter enthalten sein.

Beispiel:

```
(filter(e.sev = 5) union filter(e.sev =4))  
(filter(e.sev = 5 or e.sev =4))
```

HINWEIS: Dieser Teil ist optional. Wird dieser Teil ausgelassen, entspricht er „filter(1=1)“.

Im Abschnitt „window“ können mehrere Schnittmengenfenster enthalten sein.

Beispiel:

```
(window(w.sev = e.sev,10) intersection window(w.sip = e.sip,10))
```

HINWEIS: Dieser Teil ist optional.

Im Abschnitt „trigger“ kann eine Auslöseoperation enthalten sein.

Beispiel

```
(trigger(5,10))
```

HINWEIS: Dieser Teil ist optional. Wird dieser Teil ausgelassen, verhält sich die Regel so, als würde sie mit „trigger(1,0)“ enden.

Operatoren, die gemeinsam mit Operationen Regeln bilden

Zu den Operatoren, die gemeinsam mit Operationen Regeln bilden, zählen:

- [Flussoperator](#)
- [Vereinigungsoperator](#)
- [Schnittmengenoperator](#)

Der Vorrang der Operatoren für die Filter-, Fenster- und Auslöseoperation (in der Reihenfolge vom höchsten zum niedrigsten):

Operator	Bedeutung	Operatortyp	Assoziativität
flow	Ausgabesatz wird Eingabesatz	Binär	Von links nach rechts
intersection	Satz-Schnittmenge (doppelte Elemente entfernen)	Binär	Von links nach rechts
union	Satz-Vereinigung (doppelte Elemente entfernen)	Binär	Von links nach rechts

Flussoperator („flow“)

Der Ausgabe-Ereignissatz der Operation auf der linken Seite dient als Eingabe-Ereignissatz für die Operation auf der rechten Seite.

Beispiel:

```
filter(e.sev = 5) flow trigger(3, 60)
```

Die Ausgabe der Filteroperation ist die Eingabe der Auslöseoperation. Der Auslöser zählt nur Ereignisse mit einem Schweregrad von 5.

Vereinigungsoperator („union“)

Die Vereinigung des Ausgabesatzes der Operation auf der linken Seite und des Ausgabesatzes der Operation auf der rechten Seite. Der daraus entstehende Ausgabesatz enthält Ereignisse, die entweder aus dem Ausgabesatz der Operation auf der linken Seite oder dem Ausgabesatz der Operation auf der rechten Seite stammen (keine doppelten Elemente).

Beispiel:

```
filter(e.sev = 5) union filter(e.sip = 192.168.0.1)
```

ist äquivalent zu

```
filter(e.sev = 5 or e.sip = 192.168.0.1)
```

Schnittmengenoperator („intersection“)

Die Schnittmenge des Ausgabesatzes der Operation auf der linken Seite und des Ausgabesatzes der Operation auf der rechten Seite. Der daraus entstehende Ausgabesatz enthält Ereignisse, die sowohl im Ausgabesatz der Operation auf der linken Seite als auch im Ausgabesatz der Operation auf der rechten Seite enthalten sind (keine doppelten Elemente).

Beispiel:

```
filter(e.sev = 5) intersection filter(e.sip =  
192.17.16.32)
```

ist äquivalent zu

```
filter(e.sev = 5 and e.sip = 192.17.16.32)
```

Beispiel für Korrelationsregeln

In diesem Dokument finden Sie einen Satz von Korrelationsregeln, die auf Beispielregeln basieren, und die erforderlichen Voraussetzungen (Anforderungen), damit diese Regeln wirksam werden. Je nach Systemkonfiguration können die von Ihnen verwendeten Regeln abweichen.

Die Tags „e.rv50“ bis „e.rv53“ in den RuleLg-Beispielen entsprechen den in Ihren Collector-Zuordnungsdateien festgelegten Zuordnungen. Beispiel: Wenn Sie die Datei „windows_v2000_mapv*.csv“ oder „snort_v20_mapv*.csv“ öffnen, trifft Folgendes zu:

- Spalte „Kultur“ entspricht e.rv50
- Spalten „Gemeinschaft“ entsprechen e.rv51
- Spalte „Familie“ entspricht e.rv52
- Spalte „Ereignis“ entspricht e.rv53

Beispiel:

```
filter (e.rv52 = "Worm") flow trigger (3, 300)
```

Diese Regel gehört zur NIDS-Taxonomie. Wenn Sie in der Snort-Zuordnungsdatei die Spalte „Familie“ anzeigen, werden Sie mindestens 40 Instanzen mit dem Wort „Wurm“ finden. Diese Regel wird für mehr als 40 unterschiedliche Wurmangriffe ausgelöst, wenn diese innerhalb von 5 Minuten drei Mal auftreten.

Die folgenden Beispiele für Angriffstypen für Korrelationsregeln werden bereitgestellt.

- | | |
|---|--|
| ▪ Brute Force – Quelle und Ziel identisch | ▪ Microsoft– SQL-Server |
| ▪ Pufferüberlauf - Von derselben Quelle zu demselben Ziel | ▪ Microsoft - NETBIOS |
| ▪ Pufferüberlauf – Serviceunterbrechung | ▪ Microsoft - Windows Scripting |
| ▪ Denial of Service | ▪ Multiple Backdoor – verschiedene Quellen |
| ▪ Login Failures - Von beliebiger Quelle zu beliebigem Ziel | ▪ Multiple Backdoor – einzelne Quelle |
| ▪ Login Failures - Von derselben Quelle zu demselben Ziel | ▪ Trojanisches Pferd |
| ▪ Microsoft – Anonyme Anmeldung | ▪ UNIX - Apache-Webserver |
| ▪ Microsoft - Allgemeine Windows-Authentifizierung | ▪ UNIX - BIND/DNS |
| ▪ Microsoft - IE | ▪ UNIX - FTP |
| ▪ Microsoft – IIS | ▪ UNIX - UNIX allgemein |
| ▪ Microsoft – LAN-Manager-Authentifizierung | ▪ UNIX - Line Printer Daemon |
| ▪ Microsoft – MDAC | ▪ UNIX - Remote-Aufruf für Prozedur |
| ▪ Microsoft - Remotezugriff auf Registrierung | ▪ UNIX - Remote-Services |
| | ▪ UNIX - Sichere Shell |
| | ▪ UNIX - sendmail |
| | ▪ UNIX – SNMP |
| | ▪ Virenausbruch |
| | ▪ Wurmausbruch |

Pufferüberlauf-Angriff und Serviceunterbrechung

Mit dieser Regel werden potenzielle Sicherheitsverletzungen nach einem Pufferüberlauf-Angriff identifiziert. Bei dieser Regel erfolgt eine Warnmeldung innerhalb von 60 Sekunden, wenn am Zielort für einen Pufferüberlauf-Angriff der Service nach einem Angriff unterbrochen wurde. Ein Host-basierter Collector, HIDS/OS, kann ermitteln, ob eine Serviceunterbrechung vorliegt. Ein Pufferüberlauf-Angriff kann mit einem NIDS-, HIDS- oder OS-Collector ermittelt werden.

Falls ein System von diesem Pufferüberlauf-Angriff betroffen war, sollte dieses Ereignis untersucht werden.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	<p>Folgendes sollte vor der Implementierung dieser Regel definiert werden:</p> <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)▪ Host-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für HIDS- und OS-Taxonomie)	NIDS HIDS/OS

RuleLg für diese Regel

```
filter ((e.rv51 = "Service" and e.rv52 = "Stop" ) and  
        (e.st = "H")) flow window (w.dip = e.sip, filter  
        (e.rv52 = "Buffer_Overflow"), 60) flow trigger(1, 0)
```

Denial of Service-Angriff und Serviceunterbrechung

Mit dieser Regel werden potenzielle Sicherheitsverletzungen nach einem Denial of Service-Angriff identifiziert. Bei dieser Regel erfolgt eine Warnmeldung innerhalb von 60 Sekunden, wenn am Zielort für einen Denial of Service-Angriff der Service nach einem Angriff unterbrochen wurde. Die Serviceunterbrechung wird von einem Host-basierten Collector (HIDS/OS) ermittelt. Ein Pufferüberlauf-Angriff kann mit einem NIDS-, HIDS- oder OS-Collector ermittelt werden.

Falls ein System von einem Denial of Service-Angriff betroffen war, sollte dieses Ereignis untersucht werden.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	<p>Folgendes sollte vor der Implementierung dieser Regel definiert werden:</p> <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)▪ Host-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für HIDS- und OS-Taxonomie)	NIDS HIDS/OS

RuleLg für diese Regel

```
filter ((e.rv51 = "Service" and e.rv52 = "Stop" ) and  
      (e.st = "H")) flow window (w.dip = e.sip, filter  
      (e.rv52 = "DoS" ), 60) flow trigger(1, 0)
```

Ermitteln eines Virenausbruchs

Mit dieser Regel wird ermittelt, ob ein System in der Infrastruktur von einem bekannten Virus angegriffen wird.

Bei einem Virenangriff wird normalerweise ein System (oder auch mehrere Systeme) negativ beeinträchtigt, sodass entweder die System- und Anwendungsdaten vollständig neu geladen werden müssen oder der Unternehmensbestand vollständig verloren geht. Durch das Erkennen eines Virus bei der Verarbeitung können Schäden weitgehend reduziert oder verhindert werden.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
3 Mal innerhalb von 5 Minuten	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv52 = "Virus") flow trigger (3, 300)
```

Ermitteln eines Wurmausbruchs

Mit dieser Regel wird ermittelt, ob ein System in der Infrastruktur von einem bekannten Wurm angegriffen wird.

Bei einem Wurmangriff wird normalerweise ein System (oder auch mehrere Systeme) negativ beeinträchtigt, sodass entweder die System- und Anwendungsdaten vollständig neu geladen werden müssen oder der Unternehmensbestand vollständig verloren geht. Durch das Erkennen eines Wurms bei der Verarbeitung kann das Risiko für das Unternehmen deutlich reduziert werden.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
3 Mal innerhalb von 5 Minuten	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv52 = "Worm") flow trigger (3, 300)
```

Ermitteln von Trojanern

Mit dieser Regel kann ermittelt werden, ob sich auf einem System in der Infrastruktur ein Trojaner befindet.

Ein erfolgreicher Trojaner kann ein Zielsystem vollständig kompromittieren.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
3 Mal innerhalb von 5 Minuten	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)▪ Host-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für HIDS- und OS-Taxonomie)	NIDS HIDS/OS

RuleLg für diese Regel

```
filter (e.rv52 ="Trojan") flow trigger (3, 500)
```

Multiple Backdoor-Angriffsversuche von einer einzelnen Quelle

Mit dieser Regel werden verschiedene Versuche zum Einbinden oder Ausführen von Backdoor-Angriffen einer einzelnen Quelle korreliert.

Ein Backdoor-Programm wird normalerweise verwendet, um die vollständige Kontrolle über das Zielsystem zu erlangen und anschließend weitere Angriffe starten zu können.

Normalerweise werden mit dieser Regel Angriffsversuche von einem Eindringling erkannt, der nach einem infizierten System sucht bzw. versucht, ein System zu infizieren.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
5 Mal innerhalb von 2 Minuten	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)▪ Host-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für HIDS- und OS-Taxonomie)	NIDS HIDS/OS

RuleLg für diese Regel

```
filter (e.rv50 = "Attack" and e.rv52 = "Backdoor" ) flow  
trigger(5, 120, discriminator (e.sip))
```

Multiple Backdoor-Angriffsversuche von verschiedenen Quellen

Mit dieser Regel werden verschiedene Versuche zum Einbinden oder Ausführen eines koordinierten Backdoor-Angriffs von verschiedenen Systemen auf einen einzelnen Zielort korreliert.

Ein Backdoor-Programm wird normalerweise verwendet, um die vollständige Kontrolle über das Zielsystem zu erlangen und anschließend weitere Angriffe starten zu können.

Normalerweise kann mit der Regel Folgendes erkannt werden:

- das Zielsystem wurde gefährdet
- der Angreifer versucht, dass anfällige System auszunutzen
- der Angreifer versucht, seine Identität durch einen koordinierten Angriff zu verbergen
- oder der Angreifer hat erfahren, dass das Ziel anfällig gegenüber derartigen Angriffen ist. Falls dies zutrifft, kann dies ein Hinweis darauf sein, dass der Angreifer dieses Wissen von einer internen Quelle erhalten hat.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
5 Mal innerhalb von 2 Minuten	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)▪ Host-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für HIDS- und OS-Taxonomie)	NIDS HIDS/OS

RuleLg für diese Regel

```
filter (e.rv50 = "Attack" and e.rv52 = "Backdoor" ) flow
trigger( 5, 120, discriminator(e.dip))
```

Mehrere Fehler bei der Anmeldung von einer beliebigen Quelle bei einem beliebigen Ziel

Mit dieser Regel werden Fehler bei der Anmeldung bei gleichartigen Systemen ermittelt.

Fehler bei der Anmeldung bei gleichartigen Systemen oder Konten können ein Hinweis darauf sein, dass der Angreifer über Vorkenntnisse in Bezug auf das Netzwerk und die im Netzwerk befindlichen kritischen Systeme verfügt. Damit sollte ein Alarm ausgelöst werden. Je mehr Informationen ein Angreifer zur Verfügung hat, desto einfacher ist es für ihn, ein System zu finden, das er für seine Zwecke ausnutzen kann.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
5 Mal innerhalb von 2 Minuten	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter ((e.rv52 = "Access" or e.rv52 = "Brute_Force") and  
        e.rv51 = "User" and e.rv50 = "Attack") flow trigger  
        (5, 120)
```

Mehrere Fehler bei der Anmeldung von derselben Quelle bei demselben Ziel

Mit dieser Regel werden mehrere von derselben Quelle ausgehende Fehler bei der Anmeldung bei demselben Ziel ermittelt.

Fehler bei der Anmeldung bei gleichartigen Systemen oder Konten können ein Hinweis darauf sein, dass der Angreifer über Vorkenntnisse in Bezug auf das Netzwerk und die im Netzwerk befindlichen kritischen Systeme verfügt. Damit sollte ein Alarm ausgelöst werden. Je mehr Informationen ein Angreifer zur Verfügung hat, desto einfacher ist es für ihn, ein System zu finden, das er für seine Zwecke ausnutzen kann.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
3 Mal innerhalb von 5 Minuten	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter ((e.rv52 = "Access" or e.rv52 = "Brute_Force") and  
        e.rv51 = "User" and e.rv50 = "Attack") flow trigger  
        (5, 120, discriminator (e.sip, e.dip))
```

Pufferüberlauf-Angriff von derselben Quelle zu demselben Ziel

Mit dieser Regel wird ein Pufferüberlauf-Angriff von derselben Quellen-IP-Adresse zu derselben Ziel-IP-Adresse ermittelt.

Der Pufferüberlauf-Angriff ist der am häufigsten vorkommende Angriff auf Netzwerke und soll ein System außer Kraft setzen. Derartige Angriffe können nur am Eingang in das Netzwerk blockiert werden. Das Wissen in Bezug auf ein Angriffssystem kann dazu beitragen, dieses zu blockieren.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
5 Mal innerhalb von 3 Minuten	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)▪ Host-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für HIDS- und OS-Taxonomie)	NIDS HIDS/OS

RuleLg für diese Regel

```
filter (e.rv52 ="Buffer_Overflow" ) flow trigger (5, 180,  
discriminator (e.sip, e.dip))
```

Brute Force-Angriff, bei dem Quelle und Ziel identisch sind

Mithilfe dieser Regel wird ermittelt, ob ein System möglicherweise mit einem entschlüsselten Passwort kompromittiert wurde.

Ein kontinuierlicher Versuch, durch Kombinationen aus Benutzernamen und Passwörtern Zugriff zu erlangen, gefolgt von einer letztendlich erfolgreichen Anmeldung kann ein Hinweis darauf sein, dass sich ein Angreifer mithilfe eines Brute Force-Angriffs Zugriff verschafft hat. Falls ein derartiger Angriff erfolgreich ist, sollte das betreffende Konto geschlossen werden.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal innerhalb von 3 Minuten	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)▪ Host-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für HIDS- und OS-Taxonomie)	NIDS HIDS/OS

RuleLg für diese Regel

```
filter (e.rv53="Other" and rv52="Access" e.rv51 ="User"  
and e.rv50="Prob" and e.st = "H") flow window (w.dip =  
e.sip, filter (e.rv52="Brute Force" and  
e.rv50="Compromise"), 180) flow trigger(1, 180,  
discriminator(e.sip, e.dip))
```

Microsoft – Überprüfung von IIS-Angriffen (Internet Information Services)

Mit dieser Regel werden die laut SANS häufigsten 10 Angriffe auf Microsoft Internet Information Service (IIS) abgedeckt. Falls Sie Microsoft IIS ausführen, könnte Ihr System anfällig für derartige Angriffe sein.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_MS_IIS") flow trigger(1,60)
```

MDAC-Angriff (Microsoft – Microsoft Data Access Connector) – Überprüfung auf Remote Data Services-Angriff

Mit dieser Regel werden die laut SANS häufigsten 10 Angriffe auf MDAC abgedeckt. Durch die Verwendung von Microsoft-Produkten kann Ihr System anfällig gegenüber Angriffen sein. MDAC wird als Tool zur Integration von Microsoft-Produkten verwendet.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_MS_MDAC") flow trigger(1,60)
```

Microsoft– SQL Server-Angriffe – Überprüfung auf SQL Server-Angriffe

Mit dieser Regel werden die laut SANS häufigsten 10 Angriffe auf Microsoft SQL Server abgedeckt. Die Verwendung von Microsoft SQL Server kann das System anfällig für Angriffe machen. Es gibt verschiedene schwerwiegende Anfälligkeiten, über die Angreifer sensible Informationen abrufen, Datenbankinhalte alarmieren, SQL Server und Server-Hosts kompromittieren können.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_MS_SQLServer") flow trigger(1,60)
```

Microsoft – NETBIOS – Überprüfung auf Angriffe auf ungesicherte Windows-Netzwerkfreigaben

Mit dieser Regel werden die laut SANS häufigsten 10 Angriffe auf NETBIOS abgedeckt. Durch die Verwendung eines Microsoft-Netzwerks mit NETBIOS kann Ihr System anfällig gegenüber Angriffen sein. Bei NETBIOS handelt es sich um die ursprüngliche Software für Netzwerkdatenübertragungen von Microsoft. In aktuellen Microsoft-Netzwerken wird NETBIOS nicht mehr als Übermittlungsmedium verwendet.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_MS_NETBIOS") flow trigger(1,60)
```

Microsoft – Anonyme Anmeldung – Überprüfung auf Null-Sitzungs-Angriffe

Mit dieser Regel werden die laut SANS häufigsten 10 Angriffe auf Null-Sitzungen abgedeckt. Durch die Verwendung von Microsoft Null-Sitzungen kann Ihr System anfällig gegenüber Angriffen sein. Anonyme Benutzer können Informationen über das Netzwerk abrufen oder sich ohne Authentifizierung anmelden.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_MS_NullSessions") flow  
trigger(1,60)
```

Microsoft – LM-Authentifizierung (LAN Manager) – Überprüfung von Hashing-Angriffen auf schwachen LM

Mit dieser Regel werden die laut SANS häufigsten 10 Angriffe auf LM-Hashing abgedeckt. Bei LM werden schwächere Verschlüsselungsschemata als bei aktuellen Microsoft-Authentifizierungsprotokollen (NTLM und NTLMv2) verwendet und LM-Passwörter können innerhalb kurzer Zeit entschlüsselt werden.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_MS_LM") flow trigger(1,60)
```

Microsoft – Überprüfung auf Angriffe auf allgemeine Windows-Authentifizierung

Mit dieser Regel werden die laut SANS häufigsten 10 Angriffe auf Passwörter abgedeckt. Wenn schwache Passwörter ermittelt werden, sollten diese geändert werden.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_MS_WeakPasswords") flow  
trigger(1,60)
```

Microsoft – Überprüfung auf Internet Explorer-Angriffe (IE)

Mit dieser Regel werden die laut SANS häufigsten 10 Angriffe auf IE abgedeckt. Bei neueren Versionen von Microsoft ist diese Anwendung in die Benutzeroberfläche des Betriebssystems integriert. Bekannte Angriffe auf IE können zu einer Kompromittierung von Microsoft-Umgebungen (ab Version Windows 2000) führen.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_MS_IE") flow trigger(1,60)
```

Microsoft – Überprüfung auf Angriff für Remote-Zugriff auf Registrierung

Mit dieser Regel werden die laut SANS häufigsten 10 Angriffe auf die Microsoft-Registrierung abgedeckt. Bei einem Microsoft-Betriebssystem werden in der Registrierung alle systemdefinierten Variablen gespeichert. Die Möglichkeit, in der Registrierung Änderungen vorzunehmen oder Variablen zu ersetzen, kann schwerwiegende Auswirkungen auf die Funktionsweise und die Sicherheit einer Microsoft-Plattform haben.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_MS_Registry") flow trigger(1,60)
```

Microsoft – Überprüfung auf Angriffe auf Windows Scripting

Mit dieser Regel werden die laut SANS häufigsten 10 Angriffe auf Windows Scripting abgedeckt. Verschiedene Microsoft-Anwendungen wurden unter Verwendung der Programmiersprache Visual Basic erstellt. Durch die Möglichkeit, Befehle durch die Bereitstellung eines Scripts auszuführen, kann ein Angreifer den Zugriff und die Kontrolle über ein Microsoft-System erlangen.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_MS_Scripting") flow trigger(1,60)
```

UNIX - Überprüfung auf RPC-Angriff (Remote Procedure Call)

Mit dieser Regel werden die laut SANS häufigsten 10 Angriffe auf UNIX-RPC abgedeckt. Remoteaufrufe für Prozeduren (RPCs) stellen in einer UNIX-Umgebung eine Methode zum Zugreifen auf oder zum Ausführen einer Anwendung oder Datei in einem Remotesystem ohne Authentifizierung dar. Solange RPC aktiviert ist, können Remotebenutzer ohne Authentifizierung Befehle auf Ihrem System ausführen, für die normalerweise Benutzerberechtigungen erforderlich sind. Durch RPC werden Remoteangriffe möglich.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_Unix_RPC") flow trigger(1,60)
```

UNIX – Überprüfung auf Angriffe auf Apache Web Server

Mit dieser Regel werden die laut SANS häufigsten 10 Angriffe auf UNIX-Apache-Webserver abgedeckt. Apache Web Server ist eine kostenlose Anwendung zur Unterstützung von Webservern. Durch die Ausführung von Apache Web Server kann Ihr System anfällig gegenüber Angriffen sein.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_Unix_Apache") flow trigger(1,60)
```

UNIX – Überprüfung auf SSH-Angriff (Secure Shell)

Mit dieser Regel werden die laut SANS häufigsten 10 Angriffe auf UNIX-SSH abgedeckt. Aufgrund der zahlreichen Probleme mit Telnet und FTP wurde Secure Shell entwickelt, um die Datenübertragung zwischen zwei Maschinen zu verschlüsseln. Mit dieser Anwendung können Datenübertragungen oder Interaktionen mit einem Remotesystem über ein sicheres Verfahren erfolgen. In einzelnen Versionen dieser Anwendung wurden jedoch verschiedene Fehler ermittelt, durch die ein Angreifer die vollständige Kontrolle über ein Zielsystem übernehmen kann.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_Unix_SSH") flow trigger(1,60)
```

UNIX – Überprüfung auf SNMP-Angriff (Simple Network Management Protocol)

Mit dieser Regel werden die laut SANS häufigsten 10 Angriffe auf UNIX-SNMP abgedeckt. SNMP wurde ursprünglich für die Verwaltung von Knoten in einem Netzwerk konzipiert. Der Sicherheitsaspekt wurde in SNMP V 1.0 gar nicht und in SNMP V 3.0. teilweise berücksichtigt. SNMP ist daher Ziel zahlreicher Angriffe.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_Unix_SNMP") flow trigger(1,60)
```


UNIX – Überprüfung auf FTP-Angriffe (File Transfer Protocol)

Mit dieser Regel werden die laut SANS häufigsten 10 Angriffe auf UNIX-FTP abgedeckt. FTP (File Transfer Protocol) ist ein wichtiger Bestandteil der Kommunikation im Internet. Dies macht es auch zu einem erstklassigen Ziel für Angreifer, die den Zugriff auf das Internet bzw. aus dem Internet umleiten.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_Unix_FTP") flow trigger(1,60)
```

UNIX - Überprüfung auf Remote-Services-Angriff

Mit dieser Regel werden die laut SANS häufigsten 10 Angriffe auf UNIX-Remote-Services abgedeckt. Remote-Services stellen in einer UNIX-Umgebung eine Methode zum Zugreifen auf oder zum Ausführen einer Anwendung oder Datei in einem Remotesystem ohne Authentifizierung dar. Solange Remote-Services aktiviert sind, können Remotebenutzer ohne Authentifizierung Befehle auf Ihrem System ausführen, für die normalerweise Benutzerberechtigungen erforderlich sind. Damit werden Remoteangriffe auf Ihr System ermöglicht.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_Unix_RemoteServices") flow  
trigger(1,60)
```

UNIX – Überprüfung auf Line Printer Daemon-Angriff

Mit dieser Regel werden die laut SANS häufigsten 10 Angriffe auf Line Printer Daemon abgedeckt. Mit Line Printer Daemon werden unter UNIX Dateien gedruckt. Diese Anwendung wird in einer UNIX-Umgebung unter dem root-Konto ausgeführt. Viele der in dieser Anwendung enthaltenen Fehler geben einem Angreifer die Möglichkeit, die vollständige Kontrolle über die UNIX-Umgebung zu erlangen.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_Unix_LPD") flow trigger(1,60)
```

UNIX – Überprüfung auf Sendmail-Angriff

Mit dieser Regel werden die laut SANS häufigsten 10 UNIX-Angriffe auf Sendmail abgedeckt. Die Anwendung Sendmail verwendet SMTP (Simple Mail Transport Protocol). Diese Anwendung ist ein wichtiger Bestandteil der Kommunikation im Internet. Dies macht es auch zu einem erstklassigen Ziel für Angreifer, die den Zugriff auf das Internet bzw. aus dem Internet umleiten.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_Unix_SendMail") flow trigger(1,60)
```

UNIX – Überprüfung auf BIND/DNS-Angriff

Mit dieser Regel werden die laut SANS häufigsten 10 Angriffe auf UNIX-DNS abgedeckt. DNS (Domain Name Service) ist ein wichtiger Bestandteil der Kommunikation im Internet. Dies macht es auch zu einem erstklassigen Ziel für Angreifer, die den Zugriff auf das Internet bzw. aus dem Internet umleiten.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_Unix_DNS") flow trigger(1,60)
```

UNIX – Angriff auf allgemeine UNIX-Authentifizierung überprüft

Mit dieser Regel werden die laut SANS häufigsten 10 UNIX-Angriffe auf schwache Passwörter abgedeckt. Wenn schwache Passwörter ermittelt werden, sollten diese geändert werden.

Regelhäufigkeit	Regelanforderungen	Regeltaxonomie
1 Mal	Folgendes sollte vor der Implementierung dieser Regel definiert werden: <ul style="list-style-type: none">▪ Netzwerk-IDS-Plattformen, die von der Sentinel-Taxonomie konvertiert werden können (weitere Informationen siehe Tabelle für NIDS-Taxonomie)	NIDS

RuleLg für diese Regel

```
filter (e.rv53 = "Sans_Unix_WeakPasswords") flow  
trigger(1,60)
```

Taxonomietabellen

In diesem Abschnitt sind zwei Tabellen verfügbar. Es handelt sich dabei um die folgenden Tabellen:

- NIDS-Taxonomie
- HIDS- und OS-Taxonomie

In diesen Tabellen werden die unterschiedlichen Werte für e.rv50 bis e.rv53 für die angegebenen RuleLg-Beispiele angeführt.

NIDS-Taxonomietabelle

Aktion – Level1 (e.rv50)	System – Level2 (e.rv51)	Details – Level3 (e.rv52)	Ergebnisse – Level4 (e.rv53)
Angriff	Chat	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
	DNS	DoS	
		Access	Sans_Unix_DNS
		Buffer_Overflow	Sans_Unix_DNS
		Backdoor	
	Mail	Brute_Force	
		DoS	
		Access	Sans_Unix_SendMail
		Buffer_Overflow	Sans_Unix_SendMail
		Backdoor	Sans_MS_IE
		Brute_Force	
		DoS	

Aktion – Level1 (e.rv50)	System – Level2 (e.rv51)	Details – Level3 (e.rv52)	Ergebnisse – Level4 (e.rv53)
	Telnet	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	Datei	Access	Sans_Unix_FTP Sans_MS_WeakPasswords Sans_MS_NETBIOS
		Buffer_Overflow	Sans_Unix_FTP
		Backdoor	Sans_Unix_FTP
		Brute_Force	
		DoS	
	Web	Access	Sans_Unix_Apache Sans_MS_NETBIOS Sans_MS_WeakPasswords Sans_MS_IIS Sans_MS_Scripting Sans_MS_SQLServer Sans_MS_IE SANS_MS_MDAC
		Buffer_Overflow	Sans_Unix_Apache Sans_MS_IIS
		Backdoor	
		Brute_Force	Sans_MS_IIS
		DoS	Sans_Unix_Apache Sans_MS_IIS
	PC	Virus	Sans_MS_IE Sans_MS_IIS
		Skript	
		Wurm	Sans_MS_SQLServer
	Server	Trojaner	
		Access	Scan_MS_IIS Sans_MS_Registry Sans_MS_SQLServer Sans_MS_NETBIOS Sans_Unix_remoteServices Sans_Unix_RPC Sans_Unix_SSH
		Buffer_Overflow	Sans_Unix_RemoteServices Sans_Unix_WeakPasswords Sans_Unix_RPC Sans_Unix_LPD Sans_MS_SQLServer Sans_MS_MDAC Sans_MS_NETBIOS Sans_Unix_SSH
		Backdoor	Sans_Unix_RPC
		Brute_Force	Sans_MS_SQLServer

Aktion – Level1 (e.rv50)	System – Level2 (e.rv51)	Details – Level3 (e.rv52)	Ergebnisse – Level4 (e.rv53)
			Sans_MS_WeakPasswords
		DoS	
	Protokoll	IP	
		TCP	
		UDP	
		ICMP	
		HTTP	
		Route	
		Talk	
		XFS	
		SSH	
		IGMP	
		Time	
		News	
		Windows	
		RIP	
		IDS	
		SNMP	Sans_Unix_SNMP
		BGP	
	Benutzer	Access	Sans_Unix_WeakPasswords Sans_Unix_RemoteServices
		Buffer_Overflow	Sans_Unix_RemoteServices Sans_MS_NETBIOS
		Backdoor	
		Brute_Force	
		DoS	
Test	Chat		
	DNS		
	Mail		
	Datei		Sans_Unix_FTP
	Web		Sans_MS_IIS Sans_Unix_Apache
	PC		
	Server		Sans_MS_NullSessions Sans_MS_Registry
	Protokoll	IP	
		TCP	
		RIP	
		SNMP	Sans_Unix_SNMP
		SSH	
		Talk	
		Time	
		Windows	
		UDP	
		ICMP	
		DHCP	
	Absuchvorgang		

Aktion – Level1 (e.rv50)	System – Level2 (e.rv51)	Details – Level3 (e.rv52)	Ergebnisse – Level4 (e.rv53)
	Telnet		Sans_MS_LM
	Benutzer		Sans_MS_LM
	IDS		
Richtlinie	Porn		
Kompromittierung	Chat	Access	
		Buffer_Overflow	Sans_Unix_Weak_Passwords
		Backdoor	
		Brute_Force	
		DoS	
	DNS	Access	Sans_Unix_DNS
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	Mail	Access	Sans_Unix_SendMail
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	Telnet	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	Datei	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	Web	Access	Sans_Unix_Apache
		Buffer_Overflow	Sans_MS_IIS
		Backdoor	Sans_Unix_Apache
		Brute_Force	Sans_MS_Registry
		DoS	
	PC	Virus	
		Skript	
		Wurm	
		Trojaner	
	Server	Access	Sans_MS_SQLServer
		Buffer_Overflow	Sans_Unix_RPC
		Backdoor	Sans_MS_WeakPasswords
			Sans_MS_Registry
			Sans_Unix_SNMP
			Sans_Unix_WeakPasswords
		Brute_Force	
		DoS	

Aktion – Level1 (e.rv50)	System – Level2 (e.rv51)	Details – Level3 (e.rv52)	Ergebnisse – Level4 (e.rv53)
	Benutzer	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	

HIDS- und OS-Taxonomietabelle

Aktion – Level1 (e.rv50)	System – Level2 (e.rv51)	Details – Level3 (e.rv52)	Ergebnisse – Level4 (e.rv53)
Angriff	Datei	Löschen	App OS
		Ausführen	App OS
		Erstellen	App OS
		Ändern	App OS
		Zugreifen	App OS
	Service	Löschen	App OS
		Stoppen	App OS
		Starten	App OS
		Erstellen	App OS
		Zugreifen	App OS Priv Mail ID Network File System
		Buffer_Overflow	
		Backdoor	
		DoS	
	Config	Löschen	App OS
		Ändern	App OS
		Erstellen	App OS
		Aktivieren	App OS
		Zugreifen	App

Aktion – Level1 (e.rv50)	System – Level2 (e.rv51)	Details – Level3 (e.rv52)	Ergebnisse – Level4 (e.rv53)
			OS
	Benutzer	Erstellen	ID Auth Param Priv
		Ändern	ID Auth Param Priv
		Löschen	ID Auth Param Priv
		Zugreifen	Guest Priv Root Other
	Gruppe	Erstellen	Member Group
		Ändern	Member Group
		Löschen	Member Group
	System	Informationen	
		Arbeitsspeicher	
		Fehlersuche	
	Anomalie		
	Telnet	Zugreifen	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	Web	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	PC	Virus	
		Skript	
		Backdoor	
		Wurm	
		Trojaner	
	DNS	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	Mail	Access	

Aktion – Level1 (e.rv50)	System – Level2 (e.rv51)	Details – Level3 (e.rv52)	Ergebnisse – Level4 (e.rv53)
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
Test	Datei	Löschen	App OS
		Ausführen	App OS
		Erstellen	App OS
		Ändern	App OS
		Access	App OS
	Service	Löschen	App OS
		Stoppen	App OS
		Starten	App OS
		Erstellen	App OS
		Zugreifen	App OS File ID Mail Priv Network System
	Config	Löschen	App OS
		Ändern	App OS
		Erstellen	App OS
		Aktivieren	App OS
		Access	App OS
	Benutzer	Erstellen	ID Auth Param Priv
		Ändern	ID Auth Param Priv
		Löschen	ID

Aktion – Level1 (e.rv50)	System – Level2 (e.rv51)	Details – Level3 (e.rv52)	Ergebnisse – Level4 (e.rv53)
			Auth Param Priv
		Zugreifen	Guest Root Other
	Group	Erstellen	Member Group
		Ändern	Member Group
		Löschen	Member Group
	System	Informationen	
		Arbeitsspeicher	
		Fehlersuche	
	Anomalie		
	Web	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	Mail	Access	
		Buffer_Overflow	
		Backdoor	
		Brute_Force	
		DoS	
	Protokoll	IP	
		TCP	
		UDP	
		ICMP	
		HTTP	
		Route	
		Talk	
		XFS	
		SSH	
		IGMP	
		Time	
		News	
		Windows	
		RIP	
		IDS	
		SNMP	
		BGP	

Korrelationsausgabe

Die Ausgabestruktur der Correlation Engine ermöglicht das Sortieren, Filtern und Melden von Daten, die als Teil einer Beobachtungsliste oder einer Korrelationsregel generiert wurden.

Korrelationsregel-Ausgabestruktur

Die Standardausgabewerte lauten wie folgt:

- „RES“ wird auf „Correlation“ festgelegt, sofern nicht vom Benutzer festgelegt
- „SubRes“ wird auf „<Regel>.<Regelname>“ festgelegt, sofern nicht vom Benutzer festgelegt
- „Sev“ wird auf 4 festgelegt, sofern nicht vom Benutzer festgelegt
- ST (Sensortyp - C)
- EI (Regelmuster - SIP='1.2.3.4.', dann Semikolon, dann Regelschwellenwert in der Form 3-2-m (Beispiel für 3 Mal in 2 Minuten)
- RT2 (Regelname)

Übergebene Skriptparameter

Die übergebenen Skriptparameter haben Auswirkungen auf die Beobachtungslisten- und die Korrelationsregeln. Skriptparameter werden im Eingabefeld „Aktion durchführen“ auf der Registerkarte „Aktivierungskriterien“ in der Form %xyz% angegeben, wobei „xyz“ der Parametername ist. Die Namen der Parameter, die META-Tags darstellen, können entweder als Kurznamen (z. B. sip) oder als lange Namen (z. B. SourceIP) angegeben werden. Bei Parameternamen muss die Groß- und Kleinschreibung beachtet werden.

Parameter

Bei den ersten elf Parametern handelt es sich um spezielle Parameter. Es sind jedoch keine META-Tags. Sie entsprechen korrelierten Ereignissen. Die Parameter 12 bis 47 sind META-Tag-Parameter.

1. %RuleName% – Der Name der ausgelösten Regel (Format lautet regel.regelname).
2. %RuleName% - Der Typ der ausgelösten Regel. C für Korrelation W für Beobachtungsliste
3. %RuleDescription% – Die Beschreibung, die beim Erstellen der Regel eingegeben wurde.
4. %RuleSeverity% – Der Schweregrad der ausgelösten Regel.
5. %RuleResource% – Der Ressourcenname der ausgelösten Regel.
6. %RuleSubResource% – Der Teilressourcenname der ausgelösten Regel.
7. %RuleLg% – Die Regel in der Regelsprache der Correlation Engine (RuleLg).
8. %RuleCount% – Die Nummer der ausgelösten Regel.
9. %RuleDuration% – Die Dauer (in Sekunden) der ausgelösten Regel.

10. %RulePattern% – Eine Liste mit allen Tags in der Regelsprache und der Tag-Wert des letzten Ereignisses, das die Regel ausgelöst hat. Das Format lautet: `tsn1='value1='value2'tsn3'`, wobei:

- „tsn1“ der Tag-Kurzname 1 ist
- „tsn2“ der Tag-Kurzname 2 ist

Beispiel:

`sip='192.168.0.3'dip='2.168.0.2'`

11. %CorrelatedEventID% – Die Ereigniskennung des korrelierten Ereignisses, die von der ausgelösten Regel generiert wurde.

12. %MessageText% - Der Meldungstext der ausgelösten Regel.

13. %EventName% - Der Ereignisname der ausgelösten Regel.

Die restlichen Tags entsprechen dem Feld des letzten Ereignisses, das das korrelierte Ereignis ausgelöst hat:

14. %sev% - Schweregrad: Der normalisierte Schweregrad des Ereignisses (0-5).

15. %vul% - Anfälligkeit: Die Anfälligkeit des in diesem Ereignis identifizierten Bestands.

16. %crt% - Gefährlichkeit: Die Gefährlichkeit des in diesem Ereignis identifizierten Bestands.

17. %dt% - DateTime: Datum und Uhrzeit (normalisiert) des Ereignisses, wie vom Collector angegeben.

18. %sip% - SourceIP: Die Quellen-IP-Adresse, von der das Ereignis ausging.

19. %dip% - DestinationIP: Die Ziel-IP-Adresse, an die das Ereignis gerichtet war.

20. %id% - EventID: Die eindeutige Kennung (Unique identifier –UUID) für dieses Ereignis .

21. %src% - SourceID: Die eindeutige Kennung (Unique identifier –UUID) für den Sentinel-Prozess, der dieses Ereignis generiert hat.

22. %port% - WizardPort: Portbeschreibung für Sentinel-Collector.

23. %agent% - WizardCollector: Portbeschreibung für Sentinel-Collector.

24. %res% - Ressource: Der Ressourcenname.

25. %sres% - SubResource: Der Name der Teilresource.

26. %evt% - EventName: Der beschreibende Name des Ereignisses, wie vom Sensor gemeldet (oder angegeben). Beispiel: „Port Scan“ (Portabfrage).

27. %sn% - SensorName: Der Name des „höchsten Detektors“ des Ereignisses, wenn als unverarbeitete Daten empfangen. Beispiel: „FW1“ für eine Firewall.

28. %st% - SensorType: Der Bezeichner für den Sensortyp, der mit einem einzelnen Buchstaben angegeben wird (N, H, O, V, C, W). H: Host-basiert, N: Netzwerk-basiert, O: Sonstiges, V: Virenschutz, C: Korrelation und W: Beobachtungsliste.

29. %et% - EventTime: Die normalisierte Uhrzeit des Ereignisses, wie vom Sensor gemeldet; Analysierung in das Format: Y-M-D-H:M:S~AMPM24~TZ.

30. %prot% - Protokoll: Das Netzwerkprotokoll des Ereignisses.

31. %shn% - SourceHostName: Der Name des Quellenhosts, von dem das Ereignis ausging.

32. %sp% - SourcePort: Der Quellen-Port, von dem das Ereignis ausging.

33. %dhn% - DestinationHostName: Der Name des Zielhosts, an den das Ereignis gerichtet war.
34. %dp% - DestinationPort: Der Name des Ziel-Ports, an den das Ereignis gerichtet war.
35. %sun% - SourceUserName: Der Quellenbenutzername, der zum Initiieren des Ereignisses verwendet wurde. Beispiel: „jdoe“ während eines „su“-Versuchs.
36. %dun% - DestinationUserName: Der Zielbenutzername, für den ein Aktionsversuch ausgeführt wurde. Beispiel: Versuch, das Passwort von „root“ zurückzusetzen.
37. %fn% - FileName: Der Name des ausgeführten Programms oder der Datei, auf die zugegriffen wurde bzw. die geändert oder beeinträchtigt wurde. Beispiel: Der Name einer infizierten Datei oder eines infizierten Programms, die bzw. das von IDS ermittelt wurde.
38. %ei% - ExtendedInformation: Speichert zusätzliche im Collector erfasste Informationen. Werte innerhalb dieser Variablen werden durch Semikolon (;) getrennt. Beispiel: Eine Domäne für eine ID oder Dateinamen.
39. %rn% - ReporterName: Der Hostname oder die IP-Adresse des Geräts, in dem ein Ereignis protokolliert wurde oder von dem eine Benachrichtigung des Ereignisses gesendet wurde.
40. %pn% - ProductName: Gibt den Typ, Hersteller und den Produktcodenamen des Sensors an, von dem das Ereignis generiert wurde. Beispiel: Check Point FireWall=CPFW.
41. %msg% - Message: FreeForm-Nachrichtentext für das Ereignis.
42. %rt1% – Von Novell für Erweiterungen reserviert. Für die Verwendung mit Advisor (Zeichenkette).
43. %rt2% – Von Novell für Erweiterungen reserviert (Zeichenkette).
44. %ct1% – Reserviert für die Verwendung durch Kunden für kundenspezifische Daten (Zeichenkette).
45. %ct2% – Reserviert für die Verwendung durch Kunden für kundenspezifische Daten (Zeichenkette).
46. %rt3% – Von Novell für Erweiterungen reserviert (Zahl).
47. %ct3% – Reserviert für die Verwendung durch Kunden für kundenspezifische Daten (Zahl).
48. Parameter 46 – 145
 %rv1% bis %rv100%
 Es handelt sich dabei um META-Tags für das aktuelle Ereignis, die reservierte Variablen darstellen.
49. Parameter 146 – 245
 %cv1% bis %cv100%
 Es handelt sich dabei um META-Tags für das aktuelle Ereignis, die Kundenvariablen darstellen.

HINWEIS: Weitere Informationen zu Befehlen und Parametern finden Sie im Referenzhandbuch für Benutzer, „Kapitel 5 – META-Tags für Wizard und Sentinel“ sowie im Benutzerhandbuch, „Kapitel 9 – Registerkarte „Admin“, Korrelationsregeln“.

Bei der Verwendung des Befehls %all% gilt Folgendes:

- Falls ein Parameterwert leer oder null ist, lautet der Parameterwert E_NULL oder <fehlendes Tag>. Auf diese Weise sind jeweils 45 Parameter verfügbar, selbst wenn einige der Felder leer sind.
- Bei der Konfiguration der Correlation Engine zum Starten des HP OVO-Schnittstellenskripts empfiehlt es sich, den Namen des Skripts sowie das Parameter-Tag %all% anzugeben:
`esec_ovo %all%`
- Bei der Konfiguration der Correlation Engine zum Starten des BMC-Schnittstellenskripts empfiehlt es sich, den Namen des Skripts sowie den Parameter %all% anzugeben:
`bmc_interface.csh %all%`
- Bei der Konfiguration der Correlation Engine zum Senden einer Email empfiehlt es sich, den Namen des Email-Skripts sowie den Parameter %all%, die Email-Adresse und (optional) den Betreff anzugeben:
`email_interface.csh %all% <name>@<domain name> "My Subject"`
- Alle Skripts/Anwendungen, die von der Correlation Engine ausgeführt werden können, müssen sich im Verzeichnis \$ESEC_HOME/sentinel/exec (UNIX) bzw. %ESEC_HOME%\sentinel\bin (Windows) befinden.
- Standardmäßig werden KEINE Parameter von der Correlation Engine an die von ihr ausgeführten Skripts weitergegeben. Zum Weitergeben von Parametern an die Skripts müssen die oben angeführten %Tags% verwendet werden.
- Werden Parameter für ein Skript angegeben, können diese mithilfe von Anführungszeichen gruppiert werden. Einige Beispiele:
`%sip% %dip%` – werden als zwei Parameter behandelt.
`"%sip% %dip%"` – werden als einzelner Parameter behandelt.
`"Hello World" %sip%` – werden als zwei Parameter behandelt.
`"The message is %msg%"` – werden als einzelner Parameter behandelt.
`%msg%` – wird als einzelner Parameter behandelt (auch wenn die dafür eingesetzte Nachricht Leerzeichen enthält.)
`"%msg%"` – wird ebenfalls als einzelner Parameter behandelt (auch wenn die dafür eingesetzte Nachricht Leerzeichen enthält.)

8

Sentinel-Korrelations-Befehlszeilenoptionen

HINWEIS: Die Begriffe „Agent“ und „Collector“ sind austauschbar. Im Folgenden werden Agenten als „Collectors“ bezeichnet.

Befehlszeilenoptionen sollten nur von erfahrenen Benutzern verwendet werden. Normale Benutzer sollten keine Änderungen mithilfe dieser Optionen ausführen. Auf die Befehlszeilenoptionen können Sie wie folgt zugreifen:

Für UNIX:

```
$ESEC_HOME/sentinel/bin
```

Für Windows:

```
%ESEC_HOME%\sentinel\bin
```

Zum Ausführen der Befehlszeilenoptionen geben Sie Folgendes ein:

```
correlation_engine <Korrelations-Befehlszeilenoption>
```

Korrelations-Befehlszeilenoption	Beschreibung
-debug	Fehlersuchmodus (ausführliche Informationen zur Fehlersuche drucken)
-noErrorLogging	Fehlerprotokollierung im Windows-Ereignisprotokoll deaktivieren.
-ruleFile <Datei>	Textdatei angeben, in der die Regeln für die Verarbeitung durch die Correlation Engine-Instanz enthalten sind
-xmlruleFile <Datei>	XML-Konfigurationsdatei angeben, in der eine lokale Kopie der Regeln gespeichert wird, die in der Datenbank verfügbar sind. Standard: startup_correlation_rules.xml
-inputChannel <Zeichenkette>	Kommunikationsschicht-Eingangskanal für Correlation Engine angeben. Standard: ewizard_binary_event
-outputChannel <Zeichenkette>	Kommunikationsschicht-Ausgangskanal für Correlation Engine angeben. Standard: correlation_binary_event.
-outputUpdateChannel <Zeichenkette>	Kanal für aktualisierte Kommunikationsschicht-Ausgabe für Correlation Engine angeben. Standard: correlation_binary_event_update

Korrelations-Befehlszeilenoption	Beschreibung
-outputExecuteChannel <Zeichenkette>	Kanal zur Ausführung der Kommunikationsschicht-Ausgabe für Correlation Engine angeben. Standard: execute
-outputIncidentChannel <Zeichenkette>	Kanal für Kommunikationsschicht-Vorfall-Ausgabe für Correlation Engine angeben. Standard: app_incident_req
-service <Zeichenkette>	Kommunikationsservice (Konfigurationsparameter) für Correlation Engine angeben. Standard: correlation_engine
-mgmtInputChannel <Zeichenkette>	Kanal für Verwaltung der Kommunikationsschicht-Eingabe für Correlation Engine angeben. Standard: correlation_mgmt_input_channel
-mgmtOutputChannel <Zeichenkette>	Kanal zur Verwaltung der Kommunikationsschicht-Ausgabe für Correlation Engine angeben. Standard: correlation_mgmt_output_channel
-mgmtService <Zeichenkette>	Kommunikationsverwaltungsservice (Konfigurationsparameter) für Correlation Engine angeben. Standard: correlation_engine_mgmt
-configurationFile <Datei>	Datei zum Überschreiben der Standardkonfigurationsparameter zum Starten von Correlation Engine angeben. Standard: + 30 Sekunden der Sentinel Serverzeit.
-noStartupRules	Correlation Engine so konfigurieren, dass die Ausführung ohne Abrufen der in der Datenbank gespeicherten Regeln erfolgt. Mit der Option -ruleFile kann auch das Abrufen der Datenbank umgangen werden.
-dbTimeout <Zeitüberschreitung in Millisekunden>	Den Zeitüberschreitungswert zum Abrufen der in der Datenbank gespeicherten Regeln festlegen. Standard: 5.000 Millisekunden (ms)
-dbRetries <Anzahl>	Anzahl der erneuten Versuche zum Kontaktieren der Datenbank festlegen. Standard: 6
-name <Name der Engine>	Reporternamen dieser Correlation Engine-Instanz festlegen. Standard: Correlation Engine.

Korrelations-Befehlszeilenoption	Beschreibung
-affinityOneProcessor	Correlation Engine so konfigurieren, dass die Ausführung mit nur einem Prozessor erfolgt.
-useEventTime	Diese Option dient zu Testzwecken und sollte nicht verwendet werden.
-useNullOutput	Diese Option dient zu Testzwecken und sollte nicht verwendet werden.
-logFile <Dateiname>	Damit wird der Status an eine Datei weitergegeben.
-logPeriod <Sekunde>	Damit wird gesteuert, wie oft der Status in die Datei geschrieben wird.
-version	Build-Version anzeigen und beenden.
-help	Diese Hilfe anzeigen und beenden.

9

Sentinel Data Access Service

HINWEIS: Die Begriffe „Agent“ und „Collector“ sind austauschbar. Im Folgenden werden Agenten als „Collectors“ bezeichnet.

Der Data Access Service (DAS)-Prozess bildet den Persistenz-Service des Sentinel Servers und stellt eine MOM-Schnittstelle (Nachrichtenbus) für die Datenbank bereit. Er ermöglicht einen datengetriebenen Zugriff auf die Backend-Datenbank. Es wird eine XML-Anforderung von verschiedenen Sentinel Prozessen empfangen. Diese werden in eine Abfrage an die Datenbank umgewandelt. Die Ergebnisse der Datenbank werden verarbeitet und zurück in eine XML-Antwort konvertiert. Mit diesem Prozess werden Anforderungen unterstützt, um Ereignisse für Quick Query und Event Drill Down abzurufen, die Anfälligkeit von Informationen und Advisor-Informationen abzufragen und Konfigurationsinformationen zu manipulieren. Mithilfe von DAS erfolgt außerdem die Protokollierung aller vom Wizard Collector Manager empfangenen Ereignisse und Anforderungen zum Abrufen und Speichern von Konfigurationsinformationen.

DAS-Containerdateien

DAS stellt einen aus fünf Einzelprozessen bestehenden Container dar. Mit jedem Prozess werden unterschiedliche Arten von Datenbankoperationen ausgeführt. Diese Prozesse werden mithilfe der folgenden Dateien gesteuert:

- `das_binary.xml`: wird für Einfügevorgänge von Ereignissen und korrelierten Ereignissen verwendet
- `das_query.xml`: wird für alle anderen Datenbankoperationen verwendet
- `das_aggregation.xml`: wird für Erstellungsoperationen verwendet
- `das_itrac.xml`: wird zum Ausführen und Konfigurieren des Aktivitätsservices und zum Konfigurieren des Workflowservices verwendet
- `das_rt.xml`: wird zum Konfigurieren der Active Views-Funktion in der Sentinel-Steuerungskonsole verwendet

ACHTUNG: Die XML-Dateien sollten nicht manuell bearbeitet werden. Verwenden Sie zum Ändern von Werten in den XML-Dateien das Dienstprogramm zur Datenbankkonfiguration (`dbconfig`).

Jeder dieser Prozesse verfügt über eine aktuelle Protokolldatei, die sich unter `%ESEC_HOME%\Sentinel\log` oder `$ESEC_HOME/Sentinel/log` befindet. Dazu gehören:

- `das_query0.*.log` - Alle `das_query`-Protokolle
- `das_binary0.*.log` - Alle `das_binary`-Protokolle
- `das_itrac0.*.log` - Aktivitäts- und Workflowprotokolle
- `das_aggregation0.*.log` - Erstellungsprotokolle
- `das_rt0.*.log` - Active View-Protokolle

Mit den XML-Dateien wird Folgendes angegeben:

- **ConnectionManager**
 - Benutzername
 - Passwort
 - Hostname
 - Portnummer
 - Datenbank (Datenbankname)
 - Server (Oracle oder MSSQL)
 - maxConnections
 - batchSize
 - loadSize
- **DispatchManager** Gibt die Kanäle im Nachrichtenbus an, die von DAS abgehört werden sollen. Mit dem DispatchManager wird außerdem angegeben, welche Java-Klasse verwendet wird, um XML-Anforderungen in Java-Objekte umzuwandeln, und an welchen Handler das Java-Objekt zur Verarbeitung gesendet werden soll. Beispiel: Eine Ereignisabfrage wird über `esecurity.cracker.QuickQueryRequestCracker` in ein Java-Objekt umgewandelt. Das Java-Objekt wird vom Cracker an den `esecurity.event.request`-Handler gesendet. Vom Handler wird es zur Ausführung an einen der Services weitergesendet.
- Und andere Komponenten, die relevante DAS-Services bereitstellen.

Verwenden Sie das Dienstprogramm „dbconfig“ zum erneuten Konfigurieren der Datenbankverbindungseigenschaften für Windows.

Erneutes Konfigurieren der Datenbankverbindungseigenschaften

Diese Prozedur muss jeweils für die folgenden Containerdateinamen (containerFilename) ausgeführt werden:

- `das_binary.xml`
- `das_query.xml`
- `das_rt.xml`
- `das_aggregation.xml`
- `das_itrac.xml`

Erneutes Konfigurieren der Datenbankverbindungseigenschaften für Windows

HINWEIS: Die Datei mit den Protokolleigenschaften wird im Abstand von 10 Sekunden überprüft, um zu ermitteln, ob seit dem letzten Lesevorgang Änderungen erfolgt sind. Falls die Datei geändert wurde, wird die Datei mit Protokolleigenschaften erneut vom `LogManagerRefreshService` gelesen.

1. Melden Sie sich mit Administratorrechten für das Installationsverzeichnis der Datenbank an.
2. Wechseln Sie zu folgendem Verzeichnis:

Für Windows:

`%ESEC_HOME%\sentinel\config`

Für UNIX:

`$ESEC_HOME/sentinel/config`

3. Geben Sie den folgenden Befehl ein:

```
dbconfig -n <containerFilename> [-u Benutzername] [-p
    Passwort] [-h Hostname] [-t Portnummer] [-d
    Datenbank] [-s Server(MSSQL oder Oracle)] [-help]
    [-version]
```

DAS-Konfigurationsdateien

Die folgenden Dateien werden zum Konfigurieren der Protokollierung des DAS-Prozesses verwendet.

- das_query_log.prop
- das_binary_log.prop
- das_rt_log.prop
- das_itrac_log.prop
- das_aggregation_log.prop

Speicherort der Dateien:

Für Windows:

%ESEC_HOME%\sentinel\config

Für UNIX:

\$ESEC_HOME/sentinel/config

Diese Dateien enthalten Konfigurationsinformationen für den Konsolenhandler, der Nachrichten in einer Standardausgabe druckt, und den Dateihandler, der Nachrichten in eine Datei schreibt. Bei der Konfiguration der einzelnen Handler können die jeweils verfügbaren Optionen festgelegt werden. Mithilfe dieser Dateien können Sie festlegen, für welche Protokollnachrichten die Konfiguration gedruckt werden soll. Es sind folgende Ebenen verfügbar:

- OFF – Deaktiviert die gesamte Protokollierung
- SEVERE (höchster Wert) – gibt an, dass eine Komponente nicht ordnungsgemäß funktioniert oder kritische Daten verloren gegangen sind/beschädigt wurden
- WARNING – falls eine Aktion zukünftig eine fehlerhafte Ausführung einer Komponente verursachen kann oder falls nicht kritische Daten verloren gegangen sind/beschädigt wurden
- INFO – Prüfinformationen
- CONFIG
- FINE – zum Durchführen der Fehlersuche
- FINER – zum Durchführen der Fehlersuche
- FINEST (niedrigster Wert) – zum Durchführen der Fehlersuche
- ALL – Protokollierung aller Ebenen

Wenn eine Protokollierungsebene festgelegt wird, werden alle Protokollnachrichten dieser Ebene sowie von höheren Ebenen (siehe Liste oben) protokolliert. Wird beispielsweise die Ebene INFO angegeben, werden alle Nachrichten aus INFO, WARNING und SEVERE protokolliert.

Falls Sie Änderungen an den Dateien vornehmen, muss DAS neu gestartet werden, damit die Änderungen wirksam werden.

Die Protokolle werden gespeichert unter:

Für Windows:

```
%ESEC_HOME%\sentinel\log\das_query_0.*.log
%ESEC_HOME%\sentinel\log\das_binary_0.*.log
%ESEC_HOME%\sentinel\log\das_itrac_0.*.log
%ESEC_HOME%\sentinel\log\das_aggregation0.*.log
```

Für UNIX:

```
$ESEC_HOME/sentinel/log/das_query0.*.log
$ESEC_HOME/sentinel/log/das_binary0.*.log
$ESEC_HOME/sentinel/log/das_itrac_0.*.log
$ESEC_HOME/sentinel/log/das_aggregation0.*.log
```

Mit dem Stern (*) wird eine eindeutige Zahl zur Konfliktlösung und die Erstellungsnummer für rotierte Protokolle angegeben. Beispielsweise ist "das_query0.0.log" die Protokolldatei mit dem Index 0 und die erste Datei in einer rotierten Gruppe von Protokolldateien für den DAS-Prozess.

Native Datenbank-Connectors zum Einfügen von Ereignissen

Mithilfe nativer DB-Connectors ist eine erhöhte Leistung zum Einfügen von Ereignissen möglich. Die Verwendung des Connectors richtet sich dabei nach der verwendeten Datenbankplattform.

Nativer Datenbank-Connector für MS SQL

Verwenden Sie den nativen ADO.Net-Ereignisspeicher.

Konfiguration für nativen MS SQL-Connector

1. Installieren Sie .Net Framework auf dem Computer, auf dem DAS installiert ist.
2. Ändern Sie in der Datei "das_binary.xml" die Eigenschaft "insert.strategy" von EventStoreService > Persistor in:

```
esecurity.ccs.comp.event.jdbc.ADOLoadStrategy
```

Nativer Datenbank-Connector für Oracle

Verwenden Sie den nativen OCI-Ereignisspeicher. Auf dem DAS-Computer muss mindestens der Oracle-Client installiert sein.

Konfiguration für nativen Oracle-Connector

1. Erstellen Sie im Home-Verzeichnis von ESECADM die Datei „profile“. Fügen Sie in diese Datei den folgenden Text ein (passen Sie ORACLE_HOME für Ihre Installation an):

```
ORACLE_HOME=/build/home/oracle/OraHome
export ORACLE_HOME
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$ORACLE_HOME/lib
export LD_LIBRARY_PATH
```

2. Ändern Sie in der Datei „das_binary.xml“ die Eigenschaft „insert.strategy“ von EventStoreService > Persistor in:

```
esecurity.ccs.comp.event.jdbc.OCILoadStrategy
```


10

Standardbenutzerpasswörter ändern

HINWEIS: Die Begriffe „Agent“ und „Collector“ sind austauschbar. Im Folgenden werden Agenten als „Collectors“ bezeichnet.

In diesem Kapitel wird das Ändern von Passwörtern für Sentinel-Standardbenutzer beschrieben:

Authentifizierung für Oracle und MS SQL:

- esecadm
- esecapp
- esecdba
- esecrpt

Windows-Authentifizierung:

- Sentinel-Administrator
- Sentinel-DB-Anwendungsbewutzer
- Sentinel-DB-Administrator
- Sentinel Report-Bewutzer

Standardbenutzerpasswörter für Oracle- und MS SQL-Authentifizierung ändern

HINWEIS: Sie müssen über Administratorrechte verfügen, um Passwörter ändern zu können.

Ändern des esecadm-Passworts

Ändern des esecadm-Passworts

1. Melden Sie sich bei der Sentinel-Steuerungskonsole an und klicken Sie auf die Registerkarte *Admin*.
2. Öffnen Sie das Fenster *Benutzer-Manager*.
3. Doppelklicken Sie auf das Benutzerkonto „esecadm“ oder auf *Benutzerdetails*.
4. Ändern Sie das Passwort für das Konto.
5. Klicken Sie auf *OK*.

Ändern des esecapp-Passworts

Ändern des esecapp-Passworts

1. Für MS SQL: Verwenden Sie MS SQL Enterprise Manager und ändern Sie das esecapp-Passwort.
2. Für Oracle: Verwenden Sie Oracle Enterprise Manager und ändern Sie das esecapp-Passwort.

3. Aktualisieren Sie mithilfe des Dienstprogramms „dbconfig“ alle XML-Containerdateien. Dies ist erforderlich, weil in diesen XML-Dateien das (verschlüsselte) esecapp-Passwort gespeichert ist, das eine Datenbankverbindung von DAS und Advisor zulässt.

- das_binary.xml
- das_query.xml
- activity_container.xml
- workflow_container.xml
- das_rt.xml

Speicherort der XML-Containerdateien:

Für Windows:

```
%ESEC_HOME%\sentinel\config
```

Für Oracle:

```
$ESEC_HOME/sentinel/config
```

Informationen zum Dienstprogramm „dbconfig“ finden Sie im Sentinel-Referenzhandbuch, Kapitel 9 – „Sentinel Data Access Service“.

```
dbconfig -a <containerDirectory> -p <password>
```

Ändern des esecdba-Passworts

Ändern des esecdba-Passworts

1. Für MS SQL: Verwenden Sie MS SQL Enterprise Manager und ändern Sie das esecdba-Passwort.
2. Für Oracle: Verwenden Sie Oracle Enterprise Manager und ändern Sie das esecdba-Passwort.
3. Damit automatische SDM-Aufgaben weiterhin ausgeführt werden können (z. B. Partition hinzufügen oder Partition archivieren), ist es erforderlich, dbPass in der Datei „sdm.connect“ mit dem neuen esecdba-Passwort zu aktualisieren. Dies erfolgt über die SDM-Benutzerschnittstelle oder -Befehlszeile. Weitere Informationen finden Sie im Sentinel-Benutzerhandbuch, Kapitel 10 – „Sentinel Data Manager“.

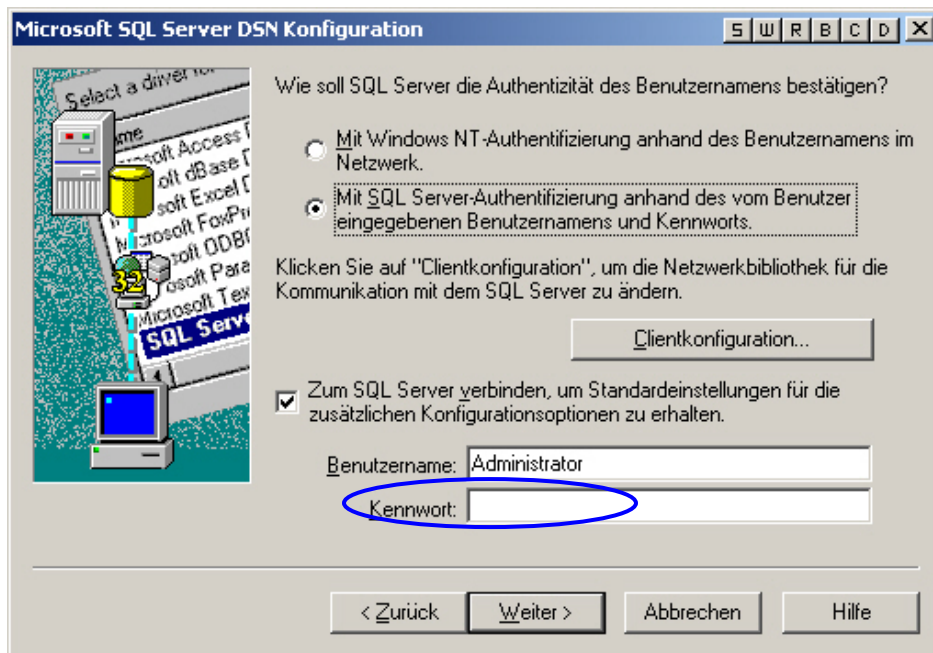
```
sdm -action saveConnection -server <oracle/mssql> -  
host <hostIp/hostname> -port <portnum> -database  
<databaseName/SID> [-driverProps <propertiesFile>]  
{-user <dbUser> -password <dbPass>} -connectFile  
<filenameToSaveConnection>
```

Ändern des esecrpt-Passworts

Ändern des esecrpt-Passworts

1. Für die Sentinel MS SQL-Datenbank: Verwenden Sie MS SQL Enterprise Manager und ändern Sie das esecrpt-Passwort.
2. Für die Sentinel Oracle-Datenbank: Verwenden Sie Oracle Enterprise Manager und ändern Sie das esecrpt-Passwort.

3. Crystal Server für Sentinel MS SQL: Aktualisieren Sie bei Bedarf den ODBC-DSN auf dem Crystal Server-Computer (*Systemsteuerung > Verwaltung > Datenquellen (ODBC)*).
 - a. Markieren Sie auf der Registerkarte „System-DSN“ den Eintrag „sentineldb“ und klicken Sie auf *Konfigurieren*.
 - b. Klicken Sie auf *Next* (Weiter). Aktualisieren Sie das Passwort.
 - c. Klicken Sie auf *Weiter*, bis die Schaltfläche *Fertig stellen* angezeigt wird. Klicken Sie auf *Fertig stellen*.



4. Crystal Server für Sentinel Oracle: Keine Änderungen erforderlich.

Standardbenutzerpasswörter für Windows-Authentifizierung ändern

Ändern des Sentinel-Administrator-Passworts

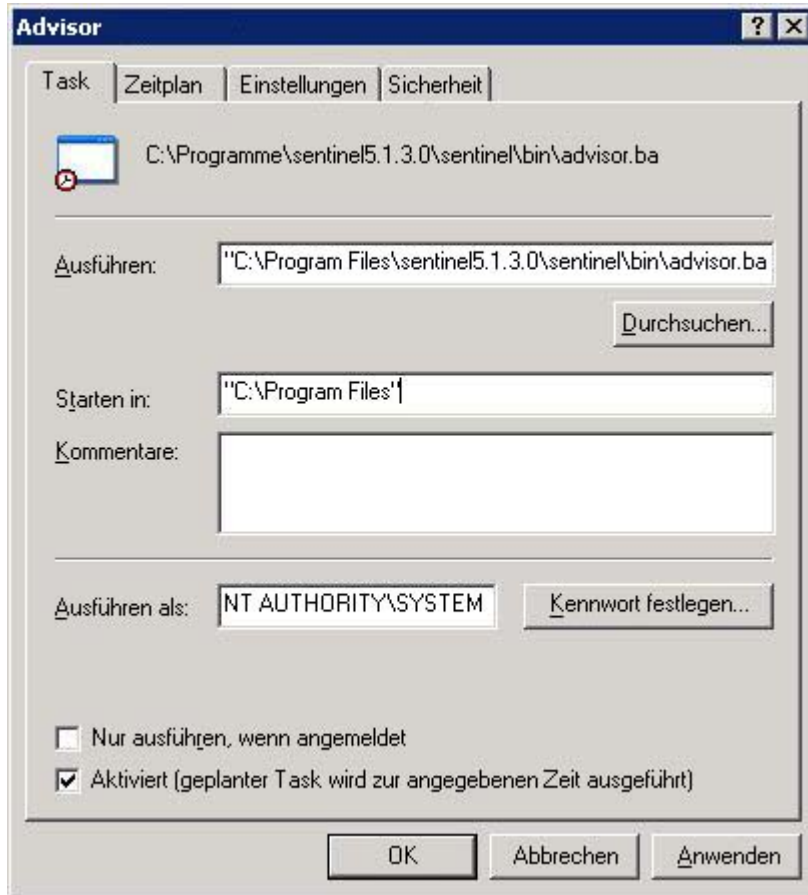
Ändern des Sentinel-Administrator-Passworts

1. Das Passwort wird im Windows-Betriebssystem geändert.

Ändern des Sentinel-DB-Administrator-Passworts

Ändern des Sentinel-DB-Administrator-Passworts

1. Das Passwort wird im Windows-Betriebssystem geändert.
2. Falls geplante SDM-Aufgaben ausgeführt werden (z. B. Partition hinzufügen oder Partition archivieren), ist es erforderlich, die Eigenschaft „Ausführen als“ (*Systemsteuerung > Geplante Tasks > mit der rechten Maustaste > Eigenschaften*) zu aktualisieren.



3. Klicken Sie auf *Passwort festlegen*. Geben Sie zweimal das neue Passwort ein und klicken Sie auf *OK*. Klicken Sie auf *Übernehmen* und dann auf *OK*.

Ändern des Administrator-Passworts für Sentinel-Anwendungs-DB

Ändern des Administrator-Passworts für Sentinel-Anwendungs-DB

1. Das Passwort wird im Windows-Betriebssystem geändert.
2. Öffnen Sie auf Ihrem DAS-Computer die Windows-Services (*Systemsteuerung > Verwaltung > Dienste*).
3. Klicken Sie mit der rechten Maustaste auf *Sentinel* und wählen Sie *Properties*. Aktivieren Sie die Registerkarte *Anmelden* und aktualisieren Sie das Passwort für *Anmelden als*. Klicken Sie auf *Übernehmen* und dann auf *OK*.

4. Falls Advisor installiert wurde, ist es erforderlich, für die geplanten Aufgaben für Advisor die Eigenschaft „Ausführen als“ zu aktualisieren (*Systemsteuerung > Geplante Tasks >* mit der rechten Maustaste auf *Eigenschaften* klicken).
5. Klicken Sie auf *Passwort festlegen*. Geben Sie zweimal das neue Passwort ein und klicken Sie auf *OK*. Klicken Sie auf *Übernehmen* und dann auf *OK*.

Ändern des Sentinel Report-Benutzer-Passworts

Ändern des Sentinel Report-Benutzer-Passworts

1. Das Passwort wird im Windows-Betriebssystem geändert.

11

Sentinel-Datenbankansichten für Oracle

HINWEIS: Die Begriffe „Agent“ und „Collector“ sind austauschbar. Im Folgenden werden Agenten als „Collectors“ bezeichnet.

In diesem Kapitel werden die Sentinel-Schemaansichten für Oracle aufgeführt. Mit diesen Ansichten erhalten Sie Informationen zum Erstellen Ihrer eigenen Berichte (Crystal Reports).

Ansichten

ADV_ALERT_CVE_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_ALERT_CVE, in der die ID-Nummer der Advisor-Warnmeldung gespeichert ist.

Spaltenname	Datentyp	Kommentar
ALERT_ID	number	Anmerkungskennung – Sequenznummer.
CVE	varchar2	
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

ADV_ALERT_PRODUCT_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_ALERT_PRODUCT, in der die Advisor-Produktinformationen (ID-Nummer für Service Pack, Version und Erstellungsdatum) gespeichert sind.

Spaltenname	Datentyp	Kommentar
ALERT_ID	number	Anmerkungskennung – Sequenznummer.
SERVICE_PACK_ID	number	
VENDOR	varchar2	
PRODUCT	varchar2	
VERSION	varchar2	Enthält die Versionsnummer
SERVICE_PACK	varchar2	
PRIMARY_FLAG	number	
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

ADV_ALERT_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_ALERT, in der die Advisor-Warmmeldungsinformationen (Name, Art der Bedrohung und Veröffentlichungsdatum) gespeichert sind.

Spaltenname	Datentyp	Kommentar
ALERT_ID	number	Anmerkungskennung – Sequenznummer.
VERSION	number	Enthält die Versionsnummer
TEMPLATE_ID	number	
TEMPLATE_NAME	varchar2	
THREAT_CATEGORY_NAME	varchar2	
THREAT_TYPE_NAME	varchar2	
HEADLINE	clob	
FIRST_PUBLISHED	date	
LAST_PUBLISHED	date	
STATUS	varchar2	
URGENCY_ID	number	
CREDIBILITY_ID	number	
SEVERITY_ID	number	
SUMMARY	clob	
LEGAL_DISCLAIMER	clob	
COPYRIGHT	varchar2	
BEGIN_EFFECTIVE_DATE	date	
END_EFFECTIVE_DATE	date	
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

ADV_ATTACK_ALERT_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_ATTACK_ALERT, in der die Advisor-Angriffsinformationen (Name, Art der Bedrohung und Veröffentlichungsdatum) gespeichert sind.

Spaltenname	Datentyp	Kommentar
ATTACK_ID	number	
ALERT_ID	number	
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

ADV_ATTACK_CVE_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_ATTACK_CVE, in der die ID der Advisor-CVE-Informationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
ATTACK_ID	number	
CVE	varchar2	
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

ADV_ATTACK_MAP_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_ATTACK_MAP, in der die ID der Advisor-Zuordnungsinformationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
ATTACK_KEY	number	
ATTACK_ID	number	
SERVICE_PACK_ID	number	
ATTACK_NAME	varchar2	
ATTACK_CODE	varchar2	
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

ADV_ATTACK_PLUGIN_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_ATTACK_PLUGIN, in der die Advisor-Plugin-Informationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
PLUGIN_KEY	number	
ATTACK_ID	number	
SERVICE_PACK_ID	number	
PLUGIN_ID	varchar2	
PLUGIN_NAME	varchar2	
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

ADV_ATTACK_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_ATTACK, in der die Advisor-Angriffsinformationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
ALERT_ID	number	
TRUSECURE_ATTACK_NAME	number	
FEED_DATE_CREATED	date	
FEED_DATE_UPDATED	date	
ATTACK_CATEGORY	varchar2	
URGENCY_ID	number	
SEVERITY_ID	number	
LOCAL	number	
REMOTE	number	
BEGIN_EFFECTIVE_DATE	date	
END_EFFECTIVE_DATE	date	
DESCRIPTION	clob	
SCENARIO	clob	
IMPACT	clob	
SAFEGUARDS	clob	
PATCHES	clob	
FALSE_POSITIVES	clob	
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

ADV_CREDIBILITY_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_CREDIBILITY, in der die Advisor-Glaubwürdigkeitsinformationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
CREDIBILITY_ID	number	
CREDIBILITY_RATING	varchar2	
CREDIBILITY_EXPLANATION	varchar2	
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

ADV_FEED_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_FEED, in der die Advisor-Feed-Informationen (Name und Datum des Feeds) gespeichert sind.

Spaltenname	Datentyp	Kommentar
FEED_NAME	varchar2	
FEED_FILE	varchar2	
BEGIN_DATE	date	
END_DATE	date	
FEED_INSERT	number	
FEED_UPDATE	number	
FEED_EXPIRE	number	

ADV_PRODUCT_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_PRODUCT, in der die Advisor-Produktinformationen (Hersteller und Produkt-ID) gespeichert sind.

Spaltenname	Datentyp	Kommentar
PRODUCT_ID	number	
VENDOR_ID	number	
PRODUCT_CATEGORY_ID	number	
PRODUCT_CATEGORY_NAME	varchar2	
PRODUCT_TYPE-ID	number	
PRODUCT_TYPE_NAME	varchar2	
PRODUCT_NAME	varchar2	
PRODUCT_DESCRIPTION	varchar2	
FEED_DATE_CREATED	date	
FEED_DATE_UPDATED	date	
ACTIVE_FLAG	number	
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

ADV_PRODUCT_SERVICE_PACK_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_PRODUCT_SERVICE_PACK, in der die Informationen zum Advisor-Service Pack (Name des Service Packs, Versionsnummer und Datum) gespeichert sind.

Spaltenname	Datentyp	Kommentar
SERVICE_PACK_ID	number	
VERSION_ID	number	Enthält die Versionsnummer
SERVICE_PACK_NAME	varchar2	
FEED_DATE_CREATED	date	
FEED_DATE_UPDATED	date	
ACTIVE_FLAG	number	
BEGIN_EFFECTIVE_DATE	date	

Spaltenname	Datentyp	Kommentar
END_EFFECTIVE_DATE	date	
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

ADV_PRODUCT_VERSION_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_PRODUCT_VERSION, in der die Advisor-Produktversionsinformationen (Versionsname, Produkt- und Versionsnummer) gespeichert sind.

Spaltenname	Datentyp	Kommentar
VERSION_ID	number	Enthält die Versionsnummer
PRODUCT_ID	number	
VERSION_NAME	varchar2	
FEED_DATE_CREATED	date	
FEED_DATE_UPDATED	date	
ACTIVE_FLAG	number	
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	number	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

ADV_SEVERITY_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_SEVERITY, in der die Advisor-Informationen zum Schweregrad-Rating gespeichert sind.

Spaltenname	Datentyp	Kommentar
SEVERITY_ID	number	
SEVERITY_RATING	varchar2	
SEVERITY_EXPLANATION	varchar2	
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

ADV_SUBALERT_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_SUBALERT.

Spaltenname	Datentyp	Kommentar
ALERT_ID	number	
SUBALERT_ID	number	
CHANGED_SECTIONS	varchar2	
VARIANTS	clob	
VIRUS_NAME	clob	
DESCRIPTION	clob	
IMPACT	clob	

Spaltenname	Datentyp	Kommentar
WARNING_INDICATORS	clob	
TECHNICAL_INFO	clob	
TRUSECURE_COMMENTS	clob	
VENDOR_ANNOUNCEMENTS	clob	
SAFEGUARDS	clob	
PATCHES_SOFTWARE	clob	
ALERT_HISTORY	clob	
BACKGROUND_INFO	clob	
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

ADV_URGENCY_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_URGENCY.

Spaltenname	Datentyp	Kommentar
URGENCY_ID	number	
URGENCY_RATING	varchar2	
URGENCY_EXPLANATION	varchar2	
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

ADV_VENDOR_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_VENDOR, in der die Advisor-Adressinformationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
VENDOR_ID	number	
VENDOR_NAME	varchar2	
CONTACT_PERSON	varchar2	
ADDRESS_LINE_1	varchar2	
ADDRESS_LINE_2	varchar2	
ADDRESS_LINE_3	varchar2	
ADDRESS_LINE_4	varchar2	
CITY	varchar2	
STATE	varchar2	
COUNTRY	varchar2	
ZIP_CODE	varchar2	
URL	varchar2	
PHONE	varchar2	
FAX	varchar2	
EMAIL	varchar2	
PAGER	varchar2	
FEED_DATE_CREATED	date	

Spaltenname	Datentyp	Kommentar
FEED_DATE_UPDATED	date	
ACTIVE_FLAG	number	
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

ADV_VULN_PRODUCT_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_VULN_PRODUCT, in der die Advisor-Anfälligkeits-Angriffs-ID und die Service Pack-ID gespeichert sind.

Spaltenname	Datentyp	Kommentar
ATTACK_ID	number	
SERVICE_PACK_ID	number	
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

ANNOTATIONS_RPT_V

Diese Ansicht verweist auf die Tabelle ANNOTATIONS, in der Dokumentation oder Hinweise gespeichert sind, die Objekten im Sentinel-System, z. B. Vorfällen, zugeordnet werden können.

Spaltenname	Datentyp	Kommentar
ANN_ID	NUMBER	Anmerkungskennung – Sequenznummer.
TEXT	VARCHAR2(4000)	Dokumentation oder Hinweise.
DATE_CREATED	DATE	Einfügedatum
DATE_MODIFIED	DATE	Datum der letzten Aktualisierung
MODIFIED_BY	NUMBER	Benutzer-ID für letzte Aktualisierung
CREATED_BY	NUMBER	Benutzer-ID beim Einfügen
ACTION	Varchar2(255)	Aktion

ASSET_CTGRY_RPT_V

Diese Ansicht verweist auf die Tabelle ASSET_CTGRY, in der Informationen zu Bestandskategorien (z. B. Hardware, Software, Betriebssystem, Datenbank usw.) gespeichert sind.

Spaltenname	Datentyp	Kommentar
ASSET_CATAGORY_ID	number	Bestandskategoriekennung
ASSET_CATAGORY_NAME	varchar2(100)	Bestandskategorienname
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

ASSET_HOSTNAME_RPT_V

Diese Ansicht verweist auf die Tabelle ASSET_HOSTNAME, in der Informationen zu alternativen Hostnamen für Bestände gespeichert sind.

Spaltenname	Datentyp	Kommentar
ASSET_HOSTNAME_ID	Varchar2(36)	Alternative Hostnamenkennung für Bestand
PHYSICAL_ASSET_ID	varchar2(36)	Kennung für physischen Bestand
HOST_NAME	Varchar2(255)	Hostname
CUSTOMER_ID	number	Kundenkennung
DATE_CREATED	date	Datum der letzten Aktualisierung
DATE_MODIFIED	date	Benutzer-ID für letzte Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

ASSET_IP_RPT_V

Diese Ansicht verweist auf die Tabelle ASSET_IP, in der Informationen zu alternativen IP-Adressen für Bestände gespeichert sind.

Spaltenname	Datentyp	Kommentar
ASSET_IP_ID	Varchar2(36)	Alternative IP-Kennung für Bestand
PHYSICAL_ASSET_ID	varchar2(36)	Kennung für physischen Bestand
IP_ADDRESS	number	IP-Adresse für Bestand
CUSTOMER_ID	number	Kundenkennung
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

ASSET_LOCATION_RPT_V

Diese Ansicht verweist auf die Tabelle ASSET_LOC, in der Informationen zu den Standorten für den Bestand gespeichert sind.

Spaltenname	Datentyp	Kommentar
LOCATION_ID	number	Standortkennung
CUSTOMER_ID	number	Kundenkennung
BUILDING_NAME	varchar2(255)	Gebäudename
ADDRESS_LINE_1	varchar2(255)	Adresszeile 1
ADDRESS_LINE_2	varchar2(255)	Adresszeile 2
CITY	varchar2(100)	Ort
STATE	varchar2(100)	Bundesland
COUNTRY	varchar2(100)	Land
ZIP_CODE	varchar2(50)	Postleitzahl
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

ASSET_RPT_V

Diese Ansicht verweist auf die Tabelle ASSET, in der Informationen zu den physischen und den immateriellen Beständen gespeichert sind.

Spaltenname	Datentyp	Kommentar
ASSET_ID	varchar2(36)	Bestandskennung
CUSTOMER_ID	number	Kundenkennung
ASSET_NAME	varchar2(255)	Bestandsname
PHYSICAL_ASSET_ID	varchar2(36)	Kennung für physischen Bestand
PRDT_ID	number	Produktkennung
ASSET_CATEGORY_ID	number	Bestandskategoriekennung
ENVIRONMENT_IDENTITY_CD	varchar2(5)	Umgebungsidentitätscode
PHYSICAL_ASSET_IND	number(1)	Indikator für physischen Bestand
ASSET_VALUE_CODE	varchar2(5)	Code für Bestandswert
CRITICALITY_CODE	varchar2(5)	Code für Bestandsgefährlichkeit
SENSITIVITY_CODE	varchar2(5)	Code für Bestandsvertraulichkeit
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

ASSET_VALUE_RPT_V

Diese Ansicht verweist auf die Tabelle ASSET_VAL_LKUP, in der Informationen zum Bestandswert gespeichert sind.

Spaltenname	Datentyp	Kommentar
ASSET_VALUE_CODE	varchar2(5)	Code für Bestandswert
ASSET_VALUE_NAME	varchar2(50)	Bestandswertname
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

ASSET_X_ENTITY_X_ROLE_RPT_V

Diese Ansicht verweist auf die Tabelle ASSET_X_ENTITY_X_ROLE, mit der ein Bestand einer Person oder einer Organisation zugewiesen wird.

Spaltenname	Datentyp	Kommentar
PERSON_ID	varchar2(36)	Personenkennung
ORGANIZATION_ID	varchar2(36)	Organisationskennung
ROLE_CODE	varchar2(5)	Funktionscode
ASSET_ID	varchar2(36)	Bestandskennung
ENTITY_TYPE_CODE	varchar2(5)	Code für Entitätstyp
PERSON_ROLE_SEQUENCE	number	Reihenfolge der Personen für eine bestimmte Funktion
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung

Spaltenname	Datentyp	Kommentar
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer der letzten Aktualisierung

ASSOCIATIONS_RPT_V

Diese Ansicht verweist auf die Tabelle ASSOCIATIONS, in der Benutzer bestimmten Vorfällen zugeordnet werden, Vorfälle zu bestimmten Anmerkungen zugeordnet werden usw.

Spaltenname	Datentyp	Kommentar
TABLE1	VARCHAR2(64)	Tabellenname 1. Beispielsweise „Vorfälle“.
ID1	VARCHAR2(36)	ID1. Beispielsweise „Vorfall-ID“.
TABLE2	VARCHAR2(64)	Tabellenname 2. Beispielsweise „Benutzer“.
ID2	VARCHAR2(36)	ID2. Beispielsweise „Benutzer-ID“.
DATE_CREATED	DATE	Einfügedatum.
DATE_MODIFIED	DATE	Datum der letzten Aktualisierung
CREATED_BY	NUMBER	Benutzer-ID beim Einfügen
MODIFIED_BY	NUMBER	Benutzer-ID für letzte Aktualisierung

ATTACHMENTS_RPT_V

Diese Ansicht verweist auf die Tabelle ATTACHMENTS, in der die Anlagendaten gespeichert sind.

Spaltenname	Datentyp	Kommentar
ATTACHMENT_ID	number	Anlagenkennung
NAME	varchar2(255)	Anlagenname
SOURCE_REFERENCE	varchar2(64)	Quellenverweis
TYPE	varchar2(32)	Anlagentyp
SUB_TYPE	varchar2(32)	Anlagenuntertyp
FILE_EXTENSION	varchar2(32)	Dateierweiterung
ATTACHMENT_DESCRIPTION	varchar2(255)	Anlagenbeschreibung
DATA	clob	Anlagendaten
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	Datum	Datum der letzten Aktualisierung
CREATED_BY	number	ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

CONFIGS_RPT_V

Diese Ansicht verweist auf die Tabelle CONFIGS, in der die Konfigurationsdaten der Anwendung gespeichert sind.

Spaltenname	Datentyp	Kommentar
USR_ID	VARCHAR2(32)	Benutzername.
APPLICATION	VARCHAR2(255)	Anwendungskennung
UNIT	VARCHAR2(64)	Anwendungseinheit
VALUE	VARCHAR2(255)	Textwert, sofern vorhanden
DATA	CLOB	XML-Daten

Spaltenname	Datentyp	Kommentar
DATE_CREATED	DATE	Einfügedatum.
DATE_MODIFIED	DATE	Datum der letzten Aktualisierung.
CREATED_BY	NUMBER	Benutzer-ID beim Einfügen.
MODIFIED_BY	NUMBER	Benutzer-ID für letzte Aktualisierung.

CONTACTS_RPT_V

Diese Ansicht verweist auf die Tabelle CONTACTS, in der die Kontaktinformationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
CNT_ID	NUMBER	Kontakt-ID – Sequenznummer
FIRST_NAME	VARCHAR2(20)	Kontakt – Vorname.
LAST_NAME	VARCHAR2(30)	Kontakt – Nachname.
TITLE	VARCHAR2(128)	Kontakt – Titel
DEPARTMENT	VARCHAR2(128)	Abteilung
PHONE	VARCHAR2(64)	Kontakt-Telefon
EMAIL	VARCHAR2(255)	Kontakt-Email-Adresse
PAGER	VARCHAR2(64)	Kontakt-Pager
CELL	VARCHAR2(64)	Kontakt-Mobiltelefon
DATE_CREATED	DATE	Einfügedatum
DATE_MODIFIED	DATE	Datum der letzten Aktualisierung
CREATED_BY	NUMBER	Benutzer-ID beim Einfügen
MODIFIED_BY	NUMBER	Benutzer-ID für letzte Aktualisierung

CORRELATED_EVENTS_RPT_V

Diese Ansicht verweist auf die Tabellen CORRELATED_EVENTS_*, in denen Informationen zu korrelierten Ereignissen gespeichert sind.

Spaltenname	Datentyp	Kommentar
PARENT_EVT_ID	varchar2	Ereignis-UUID (Universal Unique Identifier) für übergeordnetes Ereignis
CHILD_EVT_ID	varchar2	Ereignis-UUID (Universal Unique Identifier) für untergeordnetes Ereignis
PARENT_EVT_TIME	DATE	Zeit von übergeordnetem Ereignis
CHILD_EVT_TIME	DATE	Zeit von untergeordnetem Ereignis
DATE_CREATED	DATE	Von DAS erstelltes Einfügedatum
DATE_MODIFIED	DATE	Datum der letzten Aktualisierung
CREATED_BY	NUMBER	Benutzer-ID beim Einfügen
MODIFIED_BY	NUMBER	Benutzer-ID für letzte Aktualisierung

CORRELATED_EVENTS_RPT_V1

In dieser Ansicht sind aktuelle und korrelierte Verlaufsereignisse (korrelierte Ereignisse, die aus Archiven importiert wurden) enthalten.

Spaltenname	Datentyp	Kommentar
PARENT_EVT_ID	varchar2	Ereignis-UUID (Universal Unique Identifier) für übergeordnetes Ereignis

Spaltenname	Datentyp	Kommentar
CHILD_EVT_ID	varchar2	Ereignis-UUID (Universal Unique Identifier) für untergeordnetes Ereignis
PARENT_EVT_TIME	DATE	Zeit von übergeordnetem Ereignis
CHILD_EVT_TIME	DATE	Zeit von untergeordnetem Ereignis
DATE_CREATED	DATE	Von DAS erstelltes Einfügedatum
DATE_MODIFIED	DATE	Datum der letzten Aktualisierung
CREATED_BY	NUMBER	Benutzer-ID beim Einfügen
MODIFIED_BY	NUMBER	Benutzer-ID für letzte Aktualisierung

CRITICALITY_RPT_V

Diese Ansicht verweist auf die Tabelle CRIT_LKUP, in der Informationen zur Bestandsgefährlichkeit enthalten sind.

Spaltenname	Datentyp	Kommentar
CRITICALITY_CODE	varchar2(5)	Code für Bestandsgefährlichkeit
CRITICALITY_NAME	varchar2(50)	Name für Bestandsgefährlichkeit
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	Datum	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

CUST_RPT_V

Diese Ansicht verweist auf die Tabelle CUST, in der Kundeninformationen für MSSPs gespeichert sind.

Spaltenname	Datentyp	Kommentar
CUSTOMER_ID	number	Kundenkennung
CUSTOMER_NAME	varchar2(255)	Kundenname
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

ENTITY_TYPE_RPT_V

Diese Ansicht verweist auf die Tabelle ENTITY_TYP, in der Informationen zu den Entitätstypen (Person, Organisation) gespeichert sind.

Spaltenname	Datentyp	Kommentar
ENTITY_TYPE_CODE	varchar2(5)	Code für Entitätstyp
ENTITY_TYPE_NAME	varchar2(50)	Name für Entitätstyp
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	Datum	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

ENV_IDENTITY_RPT_V

Diese Ansicht verweist auf die Tabelle ENV_IDENTITY_LKUP, in der Informationen zur Umgebungsidentität für den Bestand gespeichert sind.

Spaltenname	Datentyp	Kommentar
ENVIRONMENT_IDENTITY_CODE	varchar2(5)	Umgebungsidentitätscode
ENVIRONMENT_IDENTITY_NAME	varchar2(255)	Umgebungsidentitätsname
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	Datum	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

ESEC_DISPLAY_RPT_V

Diese Ansicht verweist auf die Tabelle ESEC_DISPLAY, in der die anzeigbaren Eigenschaften von Objekten gespeichert sind. Diese werden zurzeit beim Umbenennen von META-Tags verwendet. Sie werden mit der Ereigniskonfiguration (Unternehmensrelevanz) verwendet.

Spaltenname	Datentyp	Kommentar
DISPLAY_OBJECT	VARCHAR2(32)	Das übergeordnete Objekt der Eigenschaft
TAG	VARCHAR2(32)	Der native Tagname der Eigenschaft
LABEL	VARCHAR2(32)	Die Anzeigezeichenkette des Tags.
POSITION	NUMBER	Position des Tags in der Anzeige.
WIDTH	NUMBER	Die Spaltenbreite
ALIGNMENT	NUMBER	Die horizontale Ausrichtung
FORMAT	NUMBER	Der einzeln aufgezählte Formatierer zum Anzeigen der Eigenschaft
ENABLED	VARCHAR2(1)	Gibt an, ob das Tag angezeigt wird.
TYPE	NUMBER	Gibt den Datentyp des Tags an. 1 = string 2 = ulong 3 = date 4 = uuid 5 = ipv4
DESCRIPTION	VARCHAR2(255)	Textbeschreibung des Tags
DATE_CREATED	DATE	Einfügedatum.
DATE_MODIFIED	DATE	Datum der letzten Aktualisierung.
CREATED_BY	NUMBER	Benutzer-ID beim Einfügen.
MODIFIED_BY	NUMBER	Benutzer-ID für letzte Aktualisierung.
REF_CONFIG	VARCHAR2(4000)	Referenzielle Datenkonfiguration

ESEC_PORT_REFERENCE_RPT_V

Diese Ansicht verweist auf die Tabelle ESEC_PORT_REFERENCE, in der die gemäß Branchenstandard zugewiesenen Portnummern gespeichert sind.

Spaltenname	Datentyp	Kommentar
PORT_NUMBER	NUMBER	Gemäß http://www.iana.org/assignments/port-numbers die numerische Darstellung des Ports. Diese Portnummer ist im Allgemeinen der Transportprotokollebene im TCP/IP-Stapel zugewiesen.
PROTOCOL_NUMBER	NUMBER	Gemäß http://www.iana.org/assignments/protocol-numbers die numerischen Kennungen zum Darstellen von Protokollen, die in ein IP-Paket eingekapselt sind.
PORT_KEYWORD	VARCHAR2(64)	Gemäß http://www.iana.org/assignments/port-numbers die Schlüsselwort-Darstellung des Ports.
PORT_DESCRIPTION	VARCHAR2(512)	Portbeschreibung.
DATE_CREATED	DATE	Einfügedatum.
DATE_MODIFIED	DATE	Datum der letzten Aktualisierung
CREATED_BY	NUMBER	Benutzer-ID beim Einfügen.
MODIFIED_BY	NUMBER	Benutzer-ID für letzte Änderung.

ESEC_PROTOCOL_REFERENCE_RPT_V

Diese Ansicht verweist auf die Tabelle ESEC_PROTOCOL_REFERENCE, in der die gemäß Branchenstandard zugewiesenen Protokollnummern gespeichert sind.

Spaltenname	Datentyp	Kommentar
PROTOCOL_NUMBER	NUMBER	Gemäß http://www.iana.org/assignments/protocol-numbers die numerischen Kennungen zum Darstellen von Protokollen, die in ein IP-Paket eingekapselt sind.
PROTOCOL_KEYWORD	VARCHAR2(64)	Gemäß http://www.iana.org/assignments/protocol-numbers das Schlüsselwort zum Darstellen von Protokollen, die in ein IP-Paket eingekapselt sind.
PROTOCOL_DESCRIPTION	VARCHAR2(512)	IP-Paket-Protokollbeschreibung.
DATE_CREATED	DATE	Einfügedatum.

Spaltenname	Datentyp	Kommentar
DATE_MODIFIED	DATE	Datum der letzten Aktualisierung.
CREATED_BY	NUMBER	Benutzer-ID beim Einfügen.
MODIFIED_BY	NUMBER	Benutzer-ID für letzte Aktualisierung.

ESEC_SEQUENCE _RPT_V

Diese Ansicht verweist auf die Tabelle ESEC_SEQUENCE, mit der Primärschlüssel-Sequenznummern für Sentinel-Tabellen generiert werden.

Spaltenname	Datentyp	Kommentar
TABLE_NAME	VARCHAR2(32)	Name der Tabelle.
COLUMN_NAME	VARCHAR2(32)	Name der Spalte
SEED	NUMBER	Aktueller Wert des Primärschlüsselfelds.
DATE_CREATED	DATE	Einfügedatum.
DATE_MODIFIED	DATE	Datum der letzten Aktualisierung.
CREATED_BY	NUMBER	Benutzer-ID beim Einfügen.
MODIFIED_BY	NUMBER	Benutzer-ID für letzte Aktualisierung.

EVENTS_ALL_RPT_V (aus Gründen der Abwärtskompatibilität angegeben)

In dieser Ansicht sind aktuelle und Verlaufereignisse (Ereignisse, die aus Archiven importiert wurden) enthalten.

Spaltenname	Datentyp	Kommentar
EVENT_ID	varchar2	Ereigniskennung
RESOURCE_NAME	varchar2(255)	Ressourcenname
SUB_RESOURCE	varchar2(255)	Teilressourcenname
SEVERITY	number	Ereignisschweregrad
EVENT_PARSE_TIME	date	Ereigniszeit
EVENT_DATE_TIME	date	Ereigniszeit
BASE_MESSAGE	varchar2(4000)	Basisnachricht
EVENT_NAME	varchar2(255)	Der Name des Ereignisses, wie vom Sensor gemeldet
EVENT_TIME	varchar2(255)	Zeit des Ereignisses, wie vom Sensor gemeldet
SENSOR_NAME	varchar2(255)	Sensorname
SENSOR_TYPE	varchar2(5)	Sensortyp: H – Host-basiert N – Netzwerk-basiert V – Virus O – Sonstiges
PROTOCOL	varchar2(255)	Protokollname
SOURCE-IP	number	Quellen-IP-Adresse in numerischem Format
SOURCE_HOST_NAME	varchar2(255)	Quell-Hostname
SOURCE_PORT	varchar2(32)	Quell-Port

Spaltenname	Datentyp	Kommentar
DESTINATION_IP	number	Ziel-IP-Adresse in numerischem Format
DESTINATION_HOST_NAME	varchar2(255)	Ziel-Hostname
DESTINATION_PORT	varchar2(32)	Ziel-Port
SOURCE_USER_NAME	varchar2(255)	Quellenbenutzername
DESTINATION_USER_NAME	varchar2(255)	Zielbenutzername
FILE_NAME	varchar2(1000)	Dateiname
EXTENDED_INFO	varchar2(1000)	Erweiterte Informationen
REPORT_NAME	varchar2(255)	Reportername
PRODUCT_NAME	varchar2(255)	Berichtsproduktname
CUSTOM_TAG_1	varchar2(255)	Kunden-Tag 1
CUSTOM_TAG_2	varchar2(255)	Kunden-Tag 2
CUSTOM_TAG_3	number	Kunden-Tag 3
RESERVED_TAG_1	VARCHAR2(255)	Reserviertes Tag 1 Für die zukünftige Verwendung durch Novell reserviert. Dieses Feld wird für Advisor-Informationen hinsichtlich Angriffsbeschreibungen verwendet.
RESERVED_TAG_2	varchar2(255)	Für die zukünftige Verwendung durch Novell reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RESERVED_TAG_3	number	Für die zukünftige Verwendung durch Novell reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
SOURCE_UUID	varchar(36)	Quellen-UUID
PORT	varchar(64)	Collector-Port
AGENT	varchar2(64)	Collectorname
VULNERABILITY_RATING	number	Anfälligkeits-Rating
CRITICALITY_RATING	number	Gefährlichkeits-Rating
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	Datum	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

Spaltenname	Datentyp	Kommentar
RV01–10	NUMBER	Reservierte Werte 1 bis 10 Für die zukünftige Verwendung durch Novell reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV11–20	DATE	Reservierte Werte 11 bis 20 Für die zukünftige Verwendung durch Novell reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV21–25	varchar2	Reservierte Werte 21 bis 25 Für die zukünftige Verwendung durch Novell zum Speichern von UUIDs reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV26–31	VARCHAR2(255)	Reservierte Werte 26 bis 31 Für die zukünftige Verwendung durch Novell reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV32	VARCHAR2(255)	Reservierter Wert 32 Reserviert für DeviceCategory Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV33	VARCHAR2(255)	Reservierter Wert 33 Reserviert für EventContex Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.

Spaltenname	Datentyp	Kommentar
RV34	VARCHAR2(255)	Reservierter Wert 34 Reserviert für SourceThreatLevel Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV35	VARCHAR2(255)	Reservierter Wert 35 Reserviert für SourceUserContext. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV36	VARCHAR2(255)	Reservierter Wert 36 Reserviert für DataContex. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV37	VARCHAR2(255)	Reservierter Wert 37 Reserviert für SourceFunction. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV38	VARCHAR2(255)	Reservierter Wert 38 Reserviert für SourceOperationalContext. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV39	VARCHAR2(255)	Reservierter Wert 39 Reserviert für MSSPCustomerName. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.

Spaltenname	Datentyp	Kommentar
RV40–43	VARCHAR2(255)	Reservierte Werte 40 bis 43 Für die zukünftige Verwendung durch Novell reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV44	VARCHAR2(255)	Reservierter Wert 44 Reserviert für DestinationThreatLevel. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV45	VARCHAR2(255)	Reservierter Wert 45 Reserviert für DestinationUserContext. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV46	VARCHAR2(255)	Reservierter Wert 46 Reserviert für VirusStatus. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV47	VARCHAR2(255)	Reservierter Wert 47 Für die zukünftige Verwendung durch Novell reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV48	VARCHAR2(255)	Reservierter Wert 48 Reserviert für DestinationOperationalContext. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.

Spaltenname	Datentyp	Kommentar
RV49	VARCHAR2(255)	Reservierter Wert 49 Für die zukünftige Verwendung durch Novell reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV50	VARCHAR2(255)	Taxonomiestufe 1
RV51	VARCHAR2(255)	Taxonomiestufe 2
RV52	VARCHAR2(255)	Taxonomiestufe 3
RV53	VARCHAR2(255)	Taxonomiestufe 4
CV01–10	NUMBER	Kundenwerte 1 bis 10 Reserviert für die Verwendung durch den Kunden, im Allgemeinen für die Zuordnung unternehmensrelevanter Daten
CV11–20	DATE	Kundenwerte 11 bis 20 Reserviert für die Verwendung durch den Kunden, im Allgemeinen für die Zuordnung unternehmensrelevanter Daten
CV21–100	VARCHAR2(255)	Kundenwerte 21–100 Reserviert für die Verwendung durch den Kunden, im Allgemeinen für die Zuordnung unternehmensrelevanter Daten

EVENTS_ALL_RPT_V1 (aus Gründen der Abwärtskompatibilität angegeben)

In der Ansicht sind aktuelle Ereignisse enthalten. Es werden dieselben Spalten wie für EVENT_ALL_RPT_V verwendet.

EVENTS_RPT_V (aus Gründen der Abwärtskompatibilität angegeben)

In dieser Ansicht sind aktuelle und Verlaufereignisse enthalten. Es werden dieselben Spalten wie für EVENT_ALL_RPT_V verwendet.

EVENTS_RPT_V1 (aus Gründen der Abwärtskompatibilität angegeben)

In der Ansicht sind aktuelle Ereignisse enthalten. Es werden dieselben Spalten wie für EVENT_ALL_RPT_V verwendet.

EVENTS_RPT_V2 (Alle neuen Sentinel 5-Berichte sollten diese Ansicht verwenden)

In dieser Ansicht sind aktuelle Ereignisse und Verlaufsereignisse enthalten.

Spaltenname	Datentyp	Kommentar
EVENT_ID	varchar2	Ereigniskennung
RESOURCE_NAME	varchar2(255)	Ressourcenname
SUB_RESOURCE	varchar2(255)	Teilressourcenname
SEVERITY	number	Ereignisschweregrad
EVENT_PARSE_TIME	date	Ereigniszeit
EVENT_DATETIME	date	Ereigniszeit
BASE_MESSAGE	varchar2(4000)	Basisnachricht
EVENT_NAME	varchar2(255)	Der Name des Ereignisses, wie vom Sensor gemeldet
EVENT_TIME	varchar2(255)	Zeit des Ereignisses, wie vom Sensor gemeldet
TAXONOMY_ID	number	Taxonomiekennung
PROTOCOL_ID	number	Protokollkennung
agent	number	Collectorkennung
SOURCE_IP	number	Quellen-IP-Adresse in numerischem Format
SOURCE_HOST_NAME	varchar2(255)	Quell-Hostname
SOURCE_PORT	varchar2(32)	Quell-Port
DESTINATION_IP	number	Ziel-IP-Adresse in numerischem Format
DESTINATION_HOST_NAME	varchar2(255)	Ziel-Hostname
DESTINATION_PORT	varchar2(32)	Ziel-Port
SOURCE_USER_NAME	varchar2(255)	Quellenbenutzername
DESTINATION_USER_NAME	varchar2(255)	Zielbenutzername
FILE NAME	varchar2(1000)	Dateiname
EXTENDED_INFO	varchar2(1000)	Erweiterte Informationen
CUSTOM_TAG_1	varchar2(255)	Kunden-Tag 1
CUSTOM_TAG_2	varchar2(255)	Kunden-Tag 2
CUSTOM_TAG_3	number	Kunden-Tag 3
RESERVED_TAG_1	VARCHAR2(255)	Reserviertes Tag 1 Für die zukünftige Verwendung durch Novell reserviert. Dieses Feld wird für Advisor-Informationen hinsichtlich Angriffsbeschreibungen verwendet.
RESERVED_TAG_2	varchar2(255)	Für die zukünftige Verwendung durch Novell reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.

Spaltenname	Datentyp	Kommentar
RESERVED_TAG_3	number	Für die zukünftige Verwendung durch Novell reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
VULNERABILITY_RATING	number	Anfälligkeits-Rating
CRITICALITY_RATING	number	Gefährlichkeits-Rating
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen.
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung.
RV01–10	NUMBER	Reservierte Werte 1 bis 10 Für die zukünftige Verwendung durch Novell reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV11–20	DATE	Reservierte Werte 1 bis 31 Für die zukünftige Verwendung durch Novell reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV21–25	varchar2	Reservierte Werte 21 bis 25 Für die zukünftige Verwendung durch Novell zum Speichern von UUIDs reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV26–31	VARCHAR2(255)	Reservierte Werte 26 bis 31 Für die zukünftige Verwendung durch Novell reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.

Spaltenname	Datentyp	Kommentar
RV33	VARCHAR2(255)	Reservierter Wert 33 Reserviert für EventContext Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV34	VARCHAR2(255)	Reservierter Wert 34 Reserviert für SourceThreatLevel Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV35	VARCHAR2(255)	Reservierter Wert 35 Reserviert für SourceUserContext. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV36	VARCHAR2(255)	Reservierter Wert 36 Reserviert für DataContext. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV37	VARCHAR2(255)	Reservierter Wert 37 Reserviert für SourceFunction. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV38	VARCHAR2(255)	Reservierter Wert 38 Reserviert für SourceOperationalContext. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.

Spaltenname	Datentyp	Kommentar
RV40–43	VARCHAR2(255)	Reservierte Werte 40 bis 43 Für die zukünftige Verwendung durch Novell reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV44	VARCHAR2(255)	Reservierter Wert 44 Reserviert für DestinationThreatLevel. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV45	VARCHAR2(255)	Reservierter Wert 45 Reserviert für DestinationUserContext. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV46	VARCHAR2(255)	Reservierter Wert 46 Reserviert für VirusStatus. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV47	VARCHAR2(255)	Reservierter Wert 47 Für die zukünftige Verwendung durch Novell reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV48	VARCHAR2(255)	Reservierter Wert 48 Reserviert für DestinationOperationalContext. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.

Spaltenname	Datentyp	Kommentar
RV49	VARCHAR2(255)	Reservierter Wert 49 Für die zukünftige Verwendung durch Novell reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
REFERENCE_ID 01–20	number	Für die zukünftige Verwendung durch Novell reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
CV01–10	NUMBER	Kundenwert 1 bis 10 Reserviert für die Verwendung durch den Kunden, im Allgemeinen für die Zuordnung unternehmensrelevanter Daten
CV11–20	DATE	Kundenwerte 11 bis 20 Reserviert für die Verwendung durch den Kunden, im Allgemeinen für die Zuordnung unternehmensrelevanter Daten
CV21–100	VARCHAR2(255)	Kundenwerte 21 bis 100 Reserviert für die Verwendung durch den Kunden, im Allgemeinen für die Zuordnung unternehmensrelevanter Daten

EVT_AGENT_RPT_V

Diese Ansicht verweist auf die Tabelle EVT_AGENT, in der Informationen zu den Collectors gespeichert sind.

Spaltenname	Datentyp	Kommentar
AGENT_ID	number	Collectorkennung
AGENT	varchar2(64)	Collectorname
PORT	varchar2(64)	Collector-Port
REPORT_NAME	varchar2(255)	Reportername
PRODUCT_NAME	varchar2(255)	Produktname
SENSOR_NAME	varchar2(255)	Sensorname
SENSOR_TYPE	varchar2(5)	Sensortyp: H – Host-basiert N – Netzwerkbasierend V – Virus O – Sonstiges
DEVICE_CTGRY	varchar2(255)	Geräteklasse

Spaltenname	Datentyp	Kommentar
SOURCE_UUID	varchar2	UUID (Universal Unique Identifier) der Quellkomponente
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

EVT_ASSET_RPT_V

Diese Ansicht verweist auf die Tabelle EVT_ASSET, in der Bestandsinformationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
EVENT_ASSET_ID	number	Ereignisbestandskennung
ASSET_NAME	varchar2(255)	Bestandsname
PHYSICAL_ASSET_NAME	varchar2(255)	Name für physischen Bestand
REFERENCE_ASSET_ID	varchar2(100)	Referenzbestandskennung, stellt Verknüpfung zu Quellbestands-Verwaltungssystem bereit.
MAC_ADDRESS	varchar2(100)	MAC-Adresse
RACK_NUMBER	varchar2(50)	Regalnummer
ROOM_NAME	varchar2(100)	Raumname
BUILDING_NAME	varchar2(255)	Gebäudenname
CITY	varchar2(100)	Ort
STATE	varchar2(100)	Bundesland
COUNTRY	varchar2(100)	Land
ZIP_CODE	varchar2(50)	Postleitzahl
ASSET_CATEGORY_NAME	varchar2(100)	Bestandskategorienname
NETWORK_IDENTITY_NAME	varchar2(255)	Netzwerkidentitätsname für Bestand
ENVIRONMENT_IDENTITY_NAME	varchar2(255)	Umgebungsname
ASSET_VALUE_NAME	varchar2(50)	Bestandswertname
CRITICALITY_NAME	varchar2(50)	Name für Bestandsgefährlichkeit
SENSITIVITY_NAME	varchar2(50)	Bestandsvertraulichkeitsname
CONTACT_NAME_1	varchar2(255)	Name der Kontaktperson/-organisation 1
CONTACT_NAME_2	varchar2(255)	Name der Kontaktperson/-organisation 2
ORGANIZATION_NAME_1	varchar2(100)	Bestandseigentümer-Organisationsebene 1
ORGANIZATION_NAME_2	varchar2(100)	Bestandseigentümer-Organisationsebene 2
ORGANIZATION_NAME_3	varchar2(100)	Bestandseigentümer-Organisationsebene 3
ORGANIZATION_NAME_4	varchar2(100)	Bestandseigentümer-Organisationsebene 4
DATE_CREATED	date	Einfügedatum

Spaltenname	Datentyp	Kommentar
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

EVT_DEST_EVT_NAME_SMRY_1_RPT_V

In dieser Ansicht wird die Anzahl der Ereignisse nach Ziel, Taxonomie, Ereignisname, Schweregrad und Ereigniszeit zusammengefasst.

Spaltenname	Datentyp	Kommentar
DESTINATION_IP	number	Ziel-IP-Adresse
DESTINATION_EVENT_ASSET_ID	number	Ereignisbestandskennung
TAXONOMY_ID	number	Taxonomiekennung
EVENT_NAME_ID	number	Ereignisnamenkennung
SEVERITY	number	Ereignisschweregrad
CUSTOMER_ID	number	Kundenkennung
EVT_TIME	date	Ereigniszeit
EVT_COUNT	number	Anzahl Ereignisse
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

EVT_DEST_SMRY_1_RPT_V

In dieser Ansicht sind zusammenfassende Informationen zum Ereignisziel enthalten.

Spaltenname	Datentyp	Kommentar
DESTINATION_IP	number	Ziel-IP-Adresse
DESTINATION_EVENT_ASSET_ID	number	Ereignisbestandskennung
DESTINATION_PORT	varchar2(32)	Ziel-Port
DESTINATION_USR_ID	number	Zielbenutzerkennung
TAXONOMY_ID	number	Taxonomiekennung
EVENT_NAME_ID	number	Ereignisnamenkennung
RESOURCE_ID	number	Ressourcenkennung
AGENT_ID	number	Collectorkennung
PROTOCOL_ID	number	Protokollkennung
SEVERITY	number	Ereignisschweregrad
CUSTOMER_ID	number	Kundenkennung
EVENT_TIME	date	Ereigniszeit
EVENT_CNT	number	Anzahl Ereignisse
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen

Spaltenname	Datentyp	Kommentar
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

EVT_DEST_TXNMY_SMRY_1_RPT_V

In dieser Ansicht wird die Anzahl der Ereignisse nach Ziel, Taxonomie, Schweregrad und Ereigniszeit zusammengefasst.

Spaltenname	Datentyp	Kommentar
DESTINATION_IP	number	Ziel-IP-Adresse
DESTINATION_EVENT_ASSET_ID	number	Ereignisbestandskennung
TAXONOMY_ID	number	Taxonomiekennung
SEVERITY	number	Ereignisschweregrad
CUSTOMER_ID	number	Kundenkennung
EVENT_TIME	date	Ereigniszeit
EVENT_COUNT	number	Anzahl Ereignisse
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

EVT_NAME_RPT_V

Diese Ansicht verweist auf die Tabelle EVT_NAME, in der Informationen zu Ereignisnamen gespeichert sind.

Spaltenname	Datentyp	Kommentar
EVENT_NAME_ID	number	Ereignisnamenkennung
EVENT_NAME	varchar2(255)	Ereignisname
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

EVT_PORT_SMRY_1_RPT_V

In dieser Ansicht wird die Anzahl der Ereignisse nach Ziel-Port, Schweregrad und Ereigniszeit zusammengefasst.

Spaltenname	Datentyp	Kommentar
DESTINATION_PORT	Varchar2(32)	Ziel-Port
SEVERITY	number	Ereignisschweregrad
CUSTOMER_ID	number	Kundenkennung
EVENT_TIME	date	Ereigniszeit
EVENT_COUNT	number	Anzahl Ereignisse
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung

Spaltenname	Datentyp	Kommentar
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

EVT_PRTCL_RPT_V

Diese Ansicht verweist auf die Tabelle EVT_PRTCL, in der die Ereignisprotokoll-Informationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
PROTOCOL_ID	number	Protokollkennung
PROTOCOL_NAME	varchar2(255)	Protokollname
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

EVT_RSRC_RPT_V

Diese Ansicht verweist auf die Tabelle EVT_RSRC, in der die Ereignisressourcen-Informationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
RESOURCE_ID	number	Ressourcenkennung
RESOURCE_NAME	varchar2(255)	Ressourcenname
SUBRESOURCE_NAME	varchar2(255)	Teilressourcenname
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

EVT_SEV_SMRY_1_RPT_V

In dieser Ansicht wird die Anzahl der Ereignisse nach Schweregrad und Ereigniszeit zusammengefasst.

Spaltenname	Datentyp	Kommentar
SEVERITY	number	Ereignisschweregrad
CUSTOMER_ID	number	Kundenkennung
EVENT_TIME	date	Ereigniszeit
EVENT_COUNT	number	Anzahl Ereignisse
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

EVT_SRC_SMRY_1_RPT_V

In dieser Ansicht sind zusammenfassende Informationen zu Ereignisquelle und -ziel enthalten.

Spaltenname	Datentyp	Kommentar
SOURCE_IP	number	Quellen-IP-Adresse
SOURCE_EVENT_ASSET_ID	number	Quellereignis-Bestandskennung
SOURCE_PORT	varchar2(32)	Quell-Port
SOURCE_USER_ID	number	Quellenbenutzerkennung
TAXONOMY_ID	number	Taxonomiekennung
EVENT_NAME_ID	number	Ereignisnamenkennung
RESOURCE_ID	number	Ressourcenkennung
AGENT_ID	number	Collectorkennung
PROTOCOL_ID	number	Protokollkennung
SEVERITY	number	Ereignisschweregrad
CUSTOMER_ID	number	Kundenkennung
EVENT_TIME	date	Ereigniszeit
EVENT_COUNT	number	Anzahl Ereignisse
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

EVT_TXNMY_RPT_V

Diese Ansicht verweist auf die Tabelle EVT_TXNMY, in der die Ereignistaxonomie-Informationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
TAXONOMY_ID	number	Taxonomiekennung
TAXONOMY_LEVEL_1	varchar2(100)	Taxonomiestufe 1
TAXONOMY_LEVEL_2	varchar2(100)	Taxonomiestufe 2
TAXONOMY_LEVEL_3	varchar2(100)	Taxonomiestufe 3
TAXONOMY_LEVEL_4	varchar2(100)	Taxonomiestufe 4
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

EVT_USR_RPT_V

Diese Ansicht verweist auf die Tabelle EVT_USR, in der Benutzerinformationen zu Ereignissen gespeichert sind.

Spaltenname	Datentyp	Kommentar
USER_ID	number	Benutzerkennung
USER_NAME	varchar2(255)	Benutzername
DATE_CREATED	date	Einfügedatum

Spaltenname	Datentyp	Kommentar
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

EXTERNAL_DATA_RPT_V

Diese Ansicht verweist auf die Tabelle EXTERNAL_DATA, in der externe Daten gespeichert sind.

Spaltenname	Datentyp	Kommentar
EXTERNAL_DATA_ID	number	Kennung für externe Daten
SOURCE_NAME	varchar2(50)	Quellname
SOURCE_DATA_ID	varchar2(255)	Quelldatenkennung
EXTERNAL_DATA	text	Externe Daten
EXTERNAL_DATA_TYPE	varchar2(10)	Externer Datentyp
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

HIST_EVENTS_RPT_V

In dieser Ansicht sind Verlaufereignisse (Ereignisse, die aus Archiven wiederhergestellt wurden) enthalten.

HIST_INCIDENTS_RPT_V

In dieser Ansicht sind Verlaufereignisse (Ereignisse, die aus Archiven wiederhergestellt wurden) enthalten.

IMAGES_RPT_V

Diese Ansicht verweist auf die Tabelle IMAGES, in der Bildinformationen zur Systemübersicht gespeichert sind.

Spaltenname	Datentyp	Kommentar
NAME	VARCHAR2(128)	Bildname
TYPE	VARCHAR2(64)	Bildtyp
DATA	CLOB	Bilddaten
DATE_CREATED	DATE	Einfügedatum
DATE_MODIFIED	DATE	Datum der letzten Aktualisierung
CREATED_BY	NUMBER	Benutzer-ID beim Einfügen
MODIFIED_BY	NUMBER	Benutzer-ID für letzte Aktualisierung

INCIDENTS_ASSETS_RPT_V

Diese Ansicht verweist auf die Tabelle INCIDENTS_ASSETS, in der Informationen zu den Beständen gespeichert sind, die die Vorfälle darstellen, die in der Sentinel Console erstellt werden.

Spaltenname	Datentyp	Kommentar
INC_ID	NUMBER	Vorfallkennung – Sequenznummer
ASSET_ID	varchar2	Bestands-UUID (Universal Unique Identifier)
DATE_CREATED	DATE	Einfügedatum
DATE_MODIFIED	DATE	Datum der letzten Aktualisierung
CREATED_BY	NUMBER	Benutzer-ID beim Einfügen
MODIFIED_BY	NUMBER	Benutzer-ID für letzte Aktualisierung

INCIDENTS_EVENTS_RPT_V

Diese Ansicht verweist auf die Tabelle INCIDENTS_EVENTS, in der Informationen zu den Ereignissen gespeichert sind, die die Vorfälle darstellen, die in der Sentinel Console erstellt werden.

Spaltenname	Datentyp	Kommentar
INC_ID	NUMBER	Vorfallkennung – Sequenznummer
EVT_ID	varchar2	Ereignis-UUID (Universal Unique Identifier)
EVT_TIME	DATE	Ereigniszeit
DATE_CREATED	DATE	Einfügedatum
DATE_MODIFIED	DATE	Datum der letzten Aktualisierung
CREATED_BY	NUMBER	Benutzer-ID beim Einfügen
MODIFIED_BY	NUMBER	Benutzer-ID für letzte Aktualisierung

INCIDENTS_RPT_V

Diese Ansicht verweist auf die Tabelle INCIDENTS, in der Informationen gespeichert sind, die die Details zu den Vorfällen beschreiben, die in der Sentinel Console erstellt werden.

Spaltenname	Datentyp	Kommentar
INC_ID	NUMBER	Vorfallkennung – Sequenznummer
NAME	VARCHAR2(255)	Vorfallname
SEVERITY	NUMBER	Vorfallschweregrad
STT_ID	NUMBER	Zustands-ID für Vorfall
SEVERITY_RATING	VARCHAR2(32)	Durchschnitt der Schweregrade von Ereignissen, die den Vorfall bilden.
VULNERABILITY_RATING	VARCHAR2(32)	Für die zukünftige Verwendung durch Novell reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.

Spaltenname	Datentyp	Kommentar
CRITICALITY_RATING	VARCHAR2(32)	Für die zukünftige Verwendung durch Novell reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
DATE_CREATED	DATE	Einfügedatum
DATE_MODIFIED	DATE	Datum der letzten Aktualisierung
CREATED_BY	NUMBER	Benutzer-ID beim Einfügen
MODIFIED_BY	NUMBER	Benutzer-ID für letzte Aktualisierung
INC_DESC	varchar2(4000)	Vorfallbeschreibung
INC_PRIORITY	number	Vorfallpriorität
INC_CAT	varchar2(255)	Vorfallkategorie
INC_RES	varchar2(4000)	Vorfallauflösung

INCIDENTS_VULN_RPT_V

Diese Ansicht verweist auf die Tabelle INCIDENTS_VULN, in der Informationen zu den Anfälligkeiten gespeichert sind, die die Vorfälle darstellen, die in der Sentinel Console erstellt werden.

Spaltenname	Datentyp	Kommentar
INC_ID	NUMBER	Vorfallkennung – Sequenznummer
VULN_ID	varchar2(36)	Anfälligkeits-UUID (Universal Unique Identifier)
DATE_CREATED	DATE	Einfügedatum
DATE_MODIFIED	DATE	Datum der letzten Aktualisierung
CREATED_BY	NUMBER	Benutzer-ID beim Einfügen
MODIFIED_BY	NUMBER	Benutzer-ID für letzte Aktualisierung

L_STAT_RPT_V

Diese Ansicht verweist auf die Tabelle L_STAT, in der statistische Informationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
RES_NAME	VARCHAR2(32)	Ressourcenname
STATS_NAME	VARCHAR2(32)	Name der Statistik
STATS_VALUE	VARCHAR2(32)	Wert der Statistik
OPEN_TOT_SECS	NUMERIC	Anzahl von Sekunden seit 1970.

LOGS_RPT_V

Diese Ansicht verweist auf die Tabelle LOGS_RPT, in der Protokollinformationen gespeichert sind.

PROTOKOLL-Tabelle		
Spaltenname	Datentyp	Kommentar
LOG_ID	NUMBER	Sequenznummer
TIME	DATE	Datum des Protokolls
MODULE	VARCHAR2(64)	Modulprotokoll ist für
TEXT	VARCHAR2(4000)	Protokolltext

NETWORK_IDENTITY_RPT_V

Diese Ansicht verweist auf die Tabelle NETWORK_IDENTITY_LKUP, in der Netzwerkidentitätsinformationen für den Bestand gespeichert sind.

Spaltenname	Datentyp	Kommentar
NETWORK_IDENTITY_CD	varchar2(5)	Netzwerkidentitätscode
NETWORK_IDENTITY_NAME	varchar2(255)	Netzwerkidentitätsname
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

ORGANIZATION_RPT_V

Diese Ansicht verweist auf die Tabelle ORGANIZATION, in der Informationen zur Organisation (zum Bestand) gespeichert sind.

Spaltenname	Datentyp	Kommentar
ORGANIZATION_ID	varchar2	Organisationskennung
ORGANIZATION_NAME	varchar2(100)	Name der Organisation
CUSTOMER_ID	number	Kundenkennung
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

PERSON_RPT_V

Diese Ansicht verweist auf die Tabelle PERSON, in der persönliche Informationen (Informationen zum Bestand) gespeichert sind.

Spaltenname	Datentyp	Kommentar
PERSON_ID	varchar2	Personenkennung
FIRST_NAME	varchar2(255)	Vorname
LAST_NAME	varchar2(255)	Nachname
CUSTOMER_ID	number	Kundenkennung
PHONE_NUMBER	varchar2(50)	Telefonnummer
EMAIL_ADDRESS	varchar2(255)	Email-Adresse

Spaltenname	Datentyp	Kommentar
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

PHYSICAL_ASSET_RPT_V

Diese Ansicht verweist auf die Tabelle PHYSICAL_ASSET, in der Informationen zum physischen Bestand gespeichert sind.

Spaltenname	Datentyp	Kommentar
PHYSICAL_ASSET_ID	varchar2	Kennung für physischen Bestand
CUSTOMER_ID	number	Kundenkennung
LOCATION_ID	number	Standortkennung
HOST_NAME	varchar2(255)	Hostname
IP_ADDRESS	number	IP-Adresse
NETWORK_IDENTITY_CD	varchar2(5)	Netzwerkidentitätscode
MAC_ADDRESS	varchar2(100)	MAC-Adresse
RACK_NUMBER	varchar2(50)	Regalnummer
ROOM_NAME	varchar2(100)	Raumname
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

PRODUCT_RPT_V

Diese Ansicht verweist auf die Tabelle PRDT, in der Bestands-Produktinformationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
PRODUCT_ID	number	Produktkennung
PRODUCT_NAME	varchar2(255)	Produktname
PRODUCT_VERSION	varchar2(100)	Produktversion
VENDOR_ID	number	Händlerkennung
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

ROLE_RPT_V

Diese Ansicht verweist auf die Tabelle ROLE_LKUP, in der Informationen zur Benutzerfunktion (Bestandsinformationen) gespeichert sind.

Spaltenname	Datentyp	Kommentar
ROLE_CODE	varchar2(5)	Funktionscode
ROLE_NAME	varchar2(255)	Funktionsname
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung

Spaltenname	Datentyp	Kommentar
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

SENSITIVITY_RPT_V

Diese Ansicht verweist auf die Tabelle SENSITIVITY_LKUP, in der Informationen zur Bestandsvertraulichkeit gespeichert sind.

Spaltenname	Datentyp	Kommentar
SENSITIVITY_CODE	varchar2(5)	Code für Bestandsvertraulichkeit
SENSITIVITY_NAME	varchar2(50)	Bestandsvertraulichkeitsname
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Nach Benutzer-ID
MODIFIED_BY	number	Nach Benutzer-ID

STATES_RPT_V

Diese Ansicht verweist auf die Tabelle STATES, in der Definitionen zu den Zuständen gespeichert sind, die von Anwendungen oder vom Kontext definiert werden.

Spaltenname	Datentyp	Kommentar
STT_ID	NUMBER	Zustands-ID – Sequenznummer
CONTEXT	VARCHAR2(64)	Kontext zum Zustand. Zum Beispiel Fall, Vorfall, Benutzer.
NAME	VARCHAR2(64)	Zustandsname.
TERMINAL_FLAG	VARCHAR2(1)	Gibt an, ob der Zustand des Vorfalls aufgelöst ist.
DATE_CREATED	DATE	Einfügedatum
DATE_MODIFIED	DATE	Datum der letzten Aktualisierung
MODIFIED_BY	NUMBER	Benutzer-ID beim Einfügen
CREATED_BY	NUMBER	Benutzer-ID für letzte Aktualisierung

Ansicht UNASSIGNED_INCIDENTS_RPT_V

Diese Ansicht verweist auf die Tabellen CASES und INCIDENTS, um nicht zugeordnete Fälle zu melden.

Name	Datentyp
INC_ID	NUMBER
NAME	VARCHAR2(255)
SEVERITY	NUMBER
STT_ID	NUMBER
SEVERITY_RATING	VARCHAR2(32)
VULNERABILITY_RATING	VARCHAR2(32)
CRITICALITY_RATING	VARCHAR2(32)
DATE_CREATED	DATE
DATE_MODIFIED	DATE
CREATED_BY	NUMBER
MODIFIED_BY	NUMBER

Name	Datentyp
INC_DESC	VARCHAR2(4000)
INC_PRIORITY	NUMBER
INC_CAT	VARCHAR2(255)
INC_RES	VARCHAR2(4000)

USERS_RPT_V

Diese Ansicht verweist auf die Tabelle USERS, in der alle Benutzer der Anwendung gespeichert sind. Die Benutzer werden zur Unterstützung für Drittanbieter-Berichterstellungswerkzeuge auch als Datenbankbenutzer erstellt.

Spaltenname	Datentyp	Kommentar
USR_ID	NUMBER	Benutzerkennung – Sequenznummer
NAME	VARCHAR2(64)	Kurzer, eindeutiger Benutzername, der als Anmeldename verwendet wird
CNT_ID	NUMBER	Kontakt-ID – Sequenznummer
STT_ID	NUMBER	Zustands-ID. Status ist entweder aktiv oder inaktiv.
DESCRIPTION	VARCHAR2(512)	Kommentare
DATE_CREATED	DATE	Einfügedatum
DATE_MODIFIED	DATE	Datum der letzten Aktualisierung
CREATED_BY	NUMBER	Benutzer-ID beim Einfügen
MODIFIED_BY	NUMBER	Benutzer-ID für letzte Aktualisierung
PERMISSIONS	VARCHAR2(4000)	Dem Sentinel-Benutzer aktuell zugewiesene Benutzerberechtigung
FILTER	VARCHAR2(128)	Dem Sentinel-Benutzer aktuell zugewiesener Sicherheitsfilter
UPPER_NAME	VARCHAR2(64)	Benutzername in Großschreibung
DOMAIN_AUTH_IND	NUMBER	Domänenauthentifizierungskennung

VENDOR_RPT_V

Diese Ansicht verweist auf die Tabelle VNDR, in der Informationen zu Bestandsprodukt-Händlern gespeichert sind.

Spaltenname	Datentyp	Kommentar
VENDOR_ID	number	Händlerkennung
VENDOR_NAME	varchar2(255)	Händlername
DATE_CREATED	date	Einfügedatum
DATE_MODIFIED	date	Datum der letzten Aktualisierung
CREATED_BY	number	Benutzer-ID beim Einfügen
MODIFIED_BY	number	Benutzer-ID für letzte Aktualisierung

VULN_CALC_SEVERITY_RPT_V

Diese Ansicht verweist auf die Tabellen VULN_RSRC und VULN zum Ermitteln des Schweregrad-Ratings für die Sentinel-Anfälligkeit auf der Grundlage aktueller Anfälligkeiten.

Spaltenname	Datentyp
RSRC_ID	VARCHAR2(36)
IP	VARCHAR2(32)
HOST_NAME	VARCHAR2(255)
CRITICALITY	NUMBER
ASSIGNED_VULN_SEVERITY	NUMBER
VULN_COUNT	Anzahl der Anfälligkeiten für angegebene Ressource
CALC_SEVERITY	Ermittelter Schweregrad auf Grundlage von ASSIGNED_VULN_SEVERITY und CRITICALITY

VULN_CODE_RPT_V

Diese Ansicht verweist auf die Tabelle VULN_CODE, in der von der Branche zugeordnete Anfälligkeitscodes wie CVEs und CANs von Mitre gespeichert sind.

Spaltenname	Datentyp
VULN_CODE_ID	VARCHAR2(36)
VULN_ID	VARCHAR2(36)
VULN_CODE_TYPE	VARCHAR2(64)
VULN_CODE_VALUE	VARCHAR2(255)
URL	VARCHAR2(512)
DATE_CREATED	DATE
DATE_MODIFIED	DATE
CREATED_BY	NUMBER
MODIFIED_BY	NUMBER

VULN_INFO_RPT_V

Diese Ansicht verweist auf die Tabelle VULN_INFO, in der zusätzliche Informationen gespeichert sind, die bei einem Absuchvorgang gemeldet wurden.

Spaltenname	Datentyp
VULN_INFO_ID	VARCHAR2(36)
VULN_ID	VARCHAR2(36)
VULN_INFO_TYPE	VARCHAR2(36)
VULN_INFO_VALUE	VARCHAR2(2000)
DATE_CREATED	DATE
DATE_MODIFIED	DATE
CREATED_BY	NUMBER
MODIFIED_BY	NUMBER

VULN_RPT_V

Diese Ansicht verweist auf die Tabelle VULN, in der Informationen zum abgesuchten System gespeichert sind. Jedes Absuchprogramm verfügt über einen eigenen Eintrag für jedes System.

Spaltenname	Datentyp
VULN_ID	VARCHAR2(36)
RSRC_ID	VARCHAR2(36)
PORT_NAME	VARCHAR2(64)
PORT_NUMBER	NUMBER
NETWORK_PROTOCOL	NUMBER
APPLICATION_PROTOCOL	VARCHAR2(64)
ASSIGNED_VULN_SEVERITY	NUMBER
COMPUTED_VULN_SEVERITY	NUMBER
VULN_DESCRIPTION	CLOB
VULN_SOLUTION	CLOB
VULN_SUMMARY	VARCHAR2(1000)
BEGIN_EFFECTIVE_DATE	DATE
END_EFFECTIVE_DATE	DATE
DETECTED_OS	VARCHAR2(64)
DETECTED_OS_VERSION	VARCHAR2(64)
SCANNED_APP	VARCHAR2(64)
SCANNED_APP_VERSION	VARCHAR2(64)
VULN_USER_NAME	VARCHAR2(64)
VULN_USER_DOMAIN	VARCHAR2(64)
VULN_TAXONOMY	VARCHAR2(1000)
SCANNER_CLASSIFICATION	VARCHAR2(255)
VULN_NAME	VARCHAR2(300)
VULN_MODULE	VARCHAR2(64)
DATE_CREATED	DATE
DATE_MODIFIED	DATE
CREATED_BY	NUMBER
MODIFIED_BY	NUMBER

VULN_RSRC_RPT_V

Diese Ansicht verweist auf die Tabelle VULN_RSRC, in der die einzelnen Ressourcen gespeichert sind, die für einen bestimmten Absuchvorgang abgesucht wurden.

Spaltenname	Datentyp
RSRC_ID	VARCHAR2(36)
SCANNER_ID	VARCHAR2(36)
IP	VARCHAR2(32)
HOST_NAME	VARCHAR2(255)
LOCATION	VARCHAR2(128)
DEPARTMENT	VARCHAR2(128)
BUSINESS_SYSTEM	VARCHAR2(128)
OPERATIONAL_ENVIRONMENT	VARCHAR2(64)

Spaltenname	Datentyp
CRITICALITY	NUMBER
REGULATION	VARCHAR2(128)
REGULATION_RATING	VARCHAR2(64)
DATE_CREATED	DATE
DATE_MODIFIED	DATE
CREATED_BY	NUMBER
MODIFIED_BY	NUMBER

VULN_RSRC_SCAN_RPT_V

Diese Ansicht verweist auf die Tabelle VULN_RSRC_SCAN, in der die einzelnen Ressourcen gespeichert sind, die für einen bestimmten Absuchvorgang abgesucht wurden.

Spaltenname	Datentyp
RSRC_ID	VARCHAR2(36)
SCAN_ID	VARCHAR2(36)
DATE_CREATED	DATE
DATE_MODIFIED	DATE
CREATED_BY	NUMBER
MODIFIED_BY	NUMBER

VULN_SCAN_RPT_V

Diese Ansicht verweist auf die Tabelle, in der Informationen zu Absuchvorgängen gespeichert sind.

Spaltenname	Datentyp
SCAN_ID	VARCHAR2(36)
SCANNER_ID	VARCHAR2(36)
SCAN_TYPE	VARCHAR2(10)
SCAN_START_DATE	DATE
SCAN_END_DATE	DATE
CONSOLIDATION_SERVER	VARCHAR2(64)
DATE_CREATED	DATE
DATE_MODIFIED	DATE
CREATED_BY	NUMBER
MODIFIED_BY	NUMBER

VULN_SCAN_VULN_RPT_V

Diese Ansicht verweist auf die Tabelle VULN_SCAN_VULN, in der die Anfälligkeiten gespeichert sind, die bei Absuchvorgängen ermittelt wurden.

Spaltenname	Datentyp
SCAN_ID	VARCHAR2(36)
VULN_ID	VARCHAR2(36)
DATE_CREATED	DATE
DATE_MODIFIED	DATE
CREATED_BY	NUMBER
MODIFIED_BY	NUMBER

VULN_SCANNER_RPT_V

Diese Ansicht verweist auf die Tabelle VULN_SCANNER, in der Informationen zu den Anfälligkeits-Absuchprogrammen gespeichert sind.

Spaltenname	Datentyp
SCANNER_ID	VARCHAR2(36)
PRODUCT_NAME	VARCHAR2(100)
PRODUCT_VERSION	VARCHAR2(64)
SCANNER_TYPE	VARCHAR2(64)
VENDOR	VARCHAR2(100)
SCANNER_INSTANCE	VARCHAR2(64)
DATE_CREATED	DATE
DATE_MODIFIED	DATE
CREATED_BY	NUMBER
MODIFIED_BY	NUMBER

12

Sentinel-Datenbankansichten für Microsoft SQL Server

HINWEIS: Die Begriffe „Agent“ und „Collector“ sind austauschbar. Im Folgenden werden Agenten als „Collectors“ bezeichnet.

In diesem Kapitel werden die Sentinel-Schemaansichten für Microsoft SQL Server aufgeführt. Mit diesen Ansichten erhalten Sie Informationen zum Erstellen Ihrer eigenen Berichte (Crystal Reports).

Ansichten

ADV_ALERT_CVE_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_ALERT_CVE, in der die ID der Advisor-Warmmeldung gespeichert ist.

Spaltenname	Datentyp	Kommentar
ALERT_ID	int	Anmerkungskennung - Sequenznummer.
CVE	varchar	
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Nach Benutzer-ID
MODIFIED_BY	int	Nach Benutzer-ID

ADV_ALERT_PRODUCT_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_ALERT_PRODUCT, in der die Advisor-Produktinformationen (ID-Nummer für Service Pack, Version und Erstellungsdatum) gespeichert sind.

Spaltenname	Datentyp	Kommentar
ALERT_ID	int	Anmerkungskennung – Sequenznummer.
SERVICE_PACK_ID	int	
VENDOR	varchar	
PRODUCT	varchar	
VERSION	varchar	Enthält die Versionsnummer
SERVICE_PACK	varchar	
PRIMARY_FLAG	int	
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Nach Benutzer-ID
MODIFIED_BY	int	Nach Benutzer-ID

ADV_ALERT_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_ALERT, in der die Advisor-Warnmeldungsinformationen (Name, Art der Bedrohung und Veröffentlichungsdatum) gespeichert sind.

Spaltenname	Datentyp	Kommentar
ALERT_ID	int	Anmerkungskennung - Sequenznummer.
VERSION	int	Enthält die Versionsnummer
TEMPLATE_ID	int	
TEMPLATE_NAME	varchar	
THREAT_CATEGORY_NAME	varchar	
THREAT_TYPE_NAME	varchar	
HEADLINE	text	
FIRST_PUBLISHED	datetime	
LAST_PUBLISHED	datetime	
STATUS	varchar	
URGENCY_ID	int	
CREDIBILITY_ID	int	
SEVERITY_ID	int	
SUMMARY	text	
LEGAL_DISCLAIMER	text	
COPYRIGHT	varchar	
BEGIN_EFFECTIVE_DATE	datetime	
END_EFFECTIVE_DATE	datetime	
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Nach Benutzer-ID
MODIFIED_BY	int	Nach Benutzer-ID

ADV_ATTACK_ALERT_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_ATTACK_ALERT, in der die Advisor-Angriffsinformationen (Name, Art der Bedrohung und Veröffentlichungsdatum) gespeichert sind.

Spaltenname	Datentyp	Kommentar
ATTACK_ID	int	
ALERT_ID	int	
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Nach Benutzer-ID
MODIFIED_BY	int	Nach Benutzer-ID

ADV_ATTACK_CVE_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_ATTACK_CVE, in der die ID der Advisor-CVE-Informationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
ATTACK_ID	int	
CVE	varchar	
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Nach Benutzer-ID
MODIFIED_BY	int	Nach Benutzer-ID

ADV_ATTACK_MAP_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_ATTACK_MAP, in der die ID der Advisor-Zuordnungsinformationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
ATTACK_KEY	int	
ATTACK_ID	int	
SERVICE_PACK_ID	int	
ATTACK_NAME	varchar	
ATTACK_CODE	varchar	
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Nach Benutzer-ID
MODIFIED_BY	int	Nach Benutzer-ID

ADV_ATTACK_PLUGIN_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_ATTACK_PLUGIN, in der die Advisor-Plugin-Informationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
PLUGIN_KEY	int	
ATTACK_ID	int	
SERVICE_PACK_ID	int	
PLUGIN_ID	varchar	
PLUGIN_NAME	varchar	
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Nach Benutzer-ID
MODIFIED_BY	int	Nach Benutzer-ID

ADV_ATTACK_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_ATTACK, in der die Advisor-Angriffsinformationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
ALERT_ID	int	
TRUSECURE_ATTACK_NAME	int	
FEED_DATE_CREATED	datetime	
FEED_DATE_UPDATED	datetime	
ATTACK_CATEGORY	varchar	
URGENCY_ID	int	
SEVERITY_ID	int	
LOCAL	int	
REMOTE	int	
BEGIN_EFFECTIVE_DATE	datetime	
END_EFFECTIVE_DATE	datetime	
DESCRIPTION	text	
SCENARIO	text	
IMPACT	text	
SAFEGUARDS	text	
PATCHES	text	
FALSE_POSITIVES	text	
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Nach Benutzer-ID
MODIFIED_BY	int	Nach Benutzer-ID

ADV_CREDIBILITY_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_CREDIBILITY, in der die Advisor-Glaubwürdigkeitsinformationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
CREDIBILITY_ID	int	
CREDIBILITY_RATING	varchar	
CREDIBILITY_EXPLANATION	varchar	
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Nach Benutzer-ID
MODIFIED_BY	int	Nach Benutzer-ID

ADV_FEED_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_FEED, in der die Advisor-Feed-Informationen (Name und Datum des Feeds) gespeichert sind.

Spaltenname	Datentyp	Kommentar
FEED_NAME	varchar	
FEED_FILE	varchar	
BEGIN_DATE	datetime	
END_DATE	datetime	
FEED_INSERT	int	
FEED_UPDATE	int	
FEED_EXPIRE	int	

ADV_PRODUCT_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_PRODUCT, in der die Advisor-Produktinformationen (Hersteller und Produkt-ID) gespeichert sind.

Spaltenname	Datentyp	Kommentar
PRODUCT_ID	int	
VENDOR_ID	int	
PRODUCT_CATEGORY_ID	int	
PRODUCT_CATEGORY_NAME	varchar	
PRODUCT_TYPE-ID	int	
PRODUCT_TYPE_NAME	varchar	
PRODUCT_NAME	varchar	
PRODUCT_DESCRIPTION	varchar	
FEED_DATE_CREATED	datetime	
FEED_DATE_UPDATED	datetime	
ACTIVE_FLAG	int	
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Nach Benutzer-ID
MODIFIED_BY	int	Nach Benutzer-ID

ADV_PRODUCT_SERVICE_PACK_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_PRODUCT_SERVICE_PACK, in der die Informationen zum Advisor-Service Pack (Name des Service Packs, Versionsnummer und Datum) gespeichert sind.

Spaltenname	Datentyp	Kommentar
SERVICE_PACK_ID	int	
VERSION_ID	int	Enthält die Versionsnummer
SERVICE_PACK_NAME	varchar	
FEED_DATE_CREATED	datetime	
FEED_DATE_UPDATED	datetime	
ACTIVE_FLAG	int	
BEGIN_EFFECTIVE_DATE	datetime	
END_EFFECTIVE_DATE	datetime	
DATE_CREATED	datetime	Einfügedatum

Spaltenname	Datentyp	Kommentar
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Nach Benutzer-ID
MODIFIED_BY	int	Nach Benutzer-ID

ADV_PRODUCT_VERSION_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_PRODUCT_VERSION, in der die Advisor-Produktversionsinformationen (Versionsname, Produkt- und Versionsnummer) gespeichert sind.

Spaltenname	Datentyp	Kommentar
VERSION_ID	int	Enthält die Versionsnummer
PRODUCT_ID	int	
VERSION_NAME	varchar	
FEED_DATE_CREATED	datetime	
FEED_DATE_UPDATED	datetime	
ACTIVE_FLAG	int	
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	int	Datum der letzten Aktualisierung
CREATED_BY	int	Nach Benutzer-ID
MODIFIED_BY	int	Nach Benutzer-ID

ADV_SEVERITY_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_SEVERITY, in der die Advisor-Informationen zum Schweregrad-Rating gespeichert sind.

Spaltenname	Datentyp	Kommentar
SEVERITY_ID	int	
SEVERITY_RATING	varchar	
SEVERITY_EXPLANATION	varchar	
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Nach Benutzer-ID
MODIFIED_BY	int	Nach Benutzer-ID

ADV_SUBALERT_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_SUBALERT.

Spaltenname	Datentyp	Kommentar
ALERT_ID	int	
SUBALERT_ID	int	
CHANGED_SECTIONS	varchar	
VARIANTS	text	
VIRUS_NAME	text	
DESCRIPTION	text	
IMPACT	text	
WARNING_INDICATORS	text	
TECHNICAL_INFO	text	
TRUSECURE_COMMENTS	text	

Spaltenname	Datentyp	Kommentar
VENDOR_ANNOUNCEMENTS	text	
SAFEGUARDS	text	
PATCHES_SOFTWARE	text	
ALERT_HISTORY	text	
BACKGROUND_INFO	text	
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Nach Benutzer-ID
MODIFIED_BY	int	Nach Benutzer-ID

ADV_URGENCY_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_URGENCY.

Spaltenname	Datentyp	Kommentar
URGENCY_ID	int	
URGENCY_RATING	varchar	
URGENCY_EXPLANATION	varchar	
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Nach Benutzer-ID
MODIFIED_BY	int	Nach Benutzer-ID

ADV_VENDOR_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_VENDOR, in der die Advisor-Adressinformationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
VENDOR_ID	int	
VENDOR_NAME	varchar	
CONTACT_PERSON	varchar	
ADDRESS_LINE_1	varchar	
ADDRESS_LINE_2	varchar	
ADDRESS_LINE_3	varchar	
ADDRESS_LINE_4	varchar	
CITY	varchar	
STATE	varchar	
COUNTRY	varchar	
ZIP_CODE	varchar	
URL	varchar	
PHONE	varchar	
FAX	varchar	
EMAIL	varchar	
PAGER	varchar	
FEED_DATE_CREATED	datetime	
FEED_DATE_UPDATED	datetime	
ACTIVE_FLAG	int	
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung

Spaltenname	Datentyp	Kommentar
CREATED_BY	int	Nach Benutzer-ID
MODIFIED_BY	int	Nach Benutzer-ID

ADV_VULN_PRODUCT_RPT_V

Diese Ansicht verweist auf die Tabelle ADV_VULN_PRODUCT, in der die Advisor-Anfälligkeits-Angriffs-ID und die Service Pack-ID gespeichert sind.

Spaltenname	Datentyp	Kommentar
ATTACK_ID	int	
SERVICE_PACK_ID	int	
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Nach Benutzer-ID
MODIFIED_BY	int	Nach Benutzer-ID

ANNOTATIONS_RPT_V

Diese Ansicht verweist auf die Tabelle ANNOTATIONS, in der Dokumentation oder Hinweise gespeichert sind, die Objekten im Sentinel-System, z. B. Fällen und Vorfällen, zugeordnet werden können.

Spaltenname	Datentyp	Kommentar
ANN_ID	INT	Anmerkungskennung - Sequenznummer.
TEXT	VARCHAR(4000)	Dokumentation oder Hinweise.
DATE_CREATED	DATETIME	Einfügedatum
DATE_MODIFIED	DATETIME	Datum der letzten Aktualisierung
MODIFIED_BY	INT	Benutzer-ID für letzte Aktualisierung
CREATED_BY	INT	Benutzer-ID beim Einfügen
ACTION	Varchar(255)	Aktion

ASSET_CTGRY_RPT_V

Diese Ansicht verweist auf die Tabelle ASSET_CTGRY, in der Informationen zu Bestandskategorien (z. B. Hardware, Software, Betriebssystem, Datenbank usw.) gespeichert sind.

Spaltenname	Datentyp	Kommentar
ASSET_CATEGORY_ID	bigint	Bestandskategoriekennung
ASSET_CATEGORY_NAME	varchar(100)	Bestandskategorienname
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung

ASSET_HOSTNAME_RPT_V

Diese Ansicht verweist auf die Tabelle ASSET_HOSTNAME, in der Informationen zu alternativen Hostnamen für Bestände gespeichert sind.

Spaltenname	Datentyp	Kommentar
ASSET_HOSTNAME_ID	Uniqueidentifier	Alternative Hostnamenkennung für Bestand
PHYSICAL_ASSET_ID	uniqueidentifier	Kennung für physischen Bestand
HOST_NAME	Varchar(255)	Hostname
CUSTOMER_ID	bigint	Kundenkennung
DATE_CREATED	datetime	Datum der letzten Aktualisierung
DATE_MODIFIED	datetime	Benutzer-ID für letzte Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung

ASSET_IP_RPT_V

Diese Ansicht verweist auf die Tabelle ASSET_IP, in der Informationen zu alternativen IP-Adressen für Bestände gespeichert sind.

Spaltenname	Datentyp	Kommentar
ASSET_IP_ID	Uniqueidentifier	Alternative IP-Kennung für Bestand
PHYSICAL_ASSET_ID	uniqueidentifier	Kennung für physischen Bestand
IP_ADDRESS	int	IP-Adresse für Bestand
CUSTOMER_ID	bigint	Kundenkennung
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung

ASSET_LOCATION_RPT_V

Diese Ansicht verweist auf die Tabelle ASSET_LOC, in der Informationen zu den Standorten für den Bestand gespeichert sind.

Spaltenname	Datentyp	Kommentar
LOCATION_ID	bigint	Standortkennung
CUSTOMER_ID	bigint	Kundenkennung
BUILDING_NAME	varchar(255)	Gebäudename
ADDRESS_LINE_1	varchar(255)	Adresszeile 1
ADDRESS_LINE_2	varchar(255)	Adresszeile 2
CITY	varchar(100)	Ort
STATE	varchar(100)	Status
COUNTRY	varchar(100)	Land
ZIP_CODE	varchar(50)	Postleitzahl
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung

ASSET_RPT_V

Diese Ansicht verweist auf die Tabelle ASSET, in der Informationen zu den physischen und den immateriellen Beständen gespeichert sind.

Spaltenname	Datentyp	Kommentar
ASSET_ID	uniqueidentifier	Bestandskennung
CUSTOMER_ID	bigint	Kundenkennung
ASSET_NAME	varchar(255)	Bestandsname
PHYSICAL_ASSET_ID	uniqueidentifier	Kennung für physischen Bestand
PRODUCT_ID	bigint	Produktkennung
ASSET_CATEGORY_ID	bigint	Bestandskategoriekennung
ENVIRONMENT_IDENTITY_CD	varchar(5)	Umgebungsidentitätscode
PHYSICAL_ASSET_IND	bit	Indikator für physischen Bestand
ASSET_VALUE_CD	varchar(5)	Code für Bestandswert
CRITICALITY_CODE	varchar(5)	Code für Bestandsgefährlichkeit
SENSITIVITY_CODE	varchar(5)	Code für Bestandsvertraulichkeit
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung

ASSET_VALUE_RPT_V

Diese Ansicht verweist auf die Tabelle ASSET_VAL_LKUP, in der Informationen zum Bestandswert gespeichert sind.

Spaltenname	Datentyp	Kommentar
ASSET_VALUE_CODE	varchar(5)	Code für Bestandswert
ASSET_VALUE_NAME	varchar(50)	Bestandswertname
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung

ASSET_X_ENTITY_X_ROLE_RPT_V

Diese Ansicht verweist auf die Tabelle ASSET_X_ENTITY_X_ROLE, mit der ein Bestand einer Person oder einer Organisation zugewiesen wird.

Spaltenname	Datentyp	Kommentar
PERSON_ID	uniqueidentifier	Personenkennung
ORGANIZATION_ID	uniqueidentifier	Organisationskennung
ROLE_CODE	varchar(5)	Funktionscode
ASSET_ID	uniqueidentifier	Bestandskennung
ENTITY_TYPE_CODE	varchar(5)	Code für Entitätstyp
PERSON_ROLE_SEQUENCE	int	Reihenfolge der Personen für eine bestimmte Funktion
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung

Spaltenname	Datentyp	Kommentar
CREATED_BY	int	Benutzer-ID beim Einfügen
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung

ASSOCIATIONS_RPT_V

Diese Ansicht verweist auf die Tabelle ASSOCIATIONS, in der Benutzer bestimmten Vorfällen zugeordnet werden, Vorfälle zu bestimmten Anmerkungen zugeordnet werden usw.

Spaltenname	Datentyp	Kommentar
TABLE1	VARCHAR(64)	Tabellenname 1. Beispielsweise „Vorfälle“.
ID1	VARCHAR(36)	ID1. Beispielsweise „Vorfall-ID“.
TABLE2	VARCHAR(64)	Tabellenname 2. Beispielsweise „Benutzer“.
ID2	VARCHAR(36)	ID2. Beispielsweise „Benutzer-ID“.
DATE_CREATED	DATETIME	Einfügedatum.
DATE_MODIFIED	DATETIME	Datum der letzten Aktualisierung
CREATED_BY	INT	Benutzer-ID beim Einfügen
MODIFIED_BY	INT	Benutzer-ID für letzte Aktualisierung

ATTACHMENTS_RPT_V

Diese Ansicht verweist auf die Tabelle ATTACHMENTS, in der die Anlagendaten gespeichert sind.

Spaltenname	Datentyp	Kommentar
ATTACHMENT_ID	int	Anlagenkennung
NAME	varchar(255)	Anlagenname
SOURCE_REFERENCE	varchar(64)	Quellenverweis
TYPE	varchar(32)	Anlagentyp
SUB_TYPE	varchar(32)	Anlagenuntertyp
FILE_EXTENSION	varchar(32)	Dateierweiterung
ATTACHMENT_DESCRIPTION	varchar(255)	Anlagenbeschreibung
DATA	clob	Anlagendaten
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung

CONFIGS_RPT_V

Diese Ansicht verweist auf die Tabelle CONFIGS, in der die Konfigurationsdaten der Anwendung gespeichert sind.

Spaltenname	Datentyp	Kommentar
USR_ID	VARCHAR(32)	Benutzername.
APPLICATION	VARCHAR(255)	Anwendungskennung
UNIT	VARCHAR(64)	Anwendungseinheit
VALUE	VARCHAR(255)	Textwert, sofern vorhanden
DATA	TEXT	XML-Daten

Spaltenname	Datentyp	Kommentar
DATE_CREATED	DATETIME	Einfügedatum.
DATE_MODIFIED	DATETIME	Datum der letzten Aktualisierung.
CREATED_BY	INT	Benutzer-ID beim Einfügen.
MODIFIED_BY	INT	Benutzer-ID für letzte Aktualisierung.

CONTACTS_RPT_V

Diese Ansicht verweist auf die Tabelle CONTACTS, in der die Kontaktinformationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
CNT_ID	INT	Kontakt-ID – Sequenznummer
FIRST_NAME	VARCHAR(20)	Kontakt – Vorname.
LAST_NAME	VARCHAR(30)	Kontakt – Nachname.
TITLE	VARCHAR(128)	Kontakt – Titel
DEPARTMENT	VARCHAR(128)	Department
PHONE	VARCHAR(64)	Kontakt-Telefon
EMAIL	VARCHAR(255)	Kontakt-Email-Adresse
PAGER	VARCHAR(64)	Kontakt-Pager
CELL	VARCHAR(64)	Kontakt-Mobiltelefon
DATE_CREATED	DATETIME	Einfügedatum
DATE_MODIFIED	DATETIME	Datum der letzten Aktualisierung
CREATED_BY	INT	Benutzer-ID beim Einfügen
MODIFIED_BY	INT	Benutzer-ID für letzte Aktualisierung

CORRELATED_EVENTS_RPT_V

Diese Ansicht verweist auf die Tabellen CORRELATED_EVENTS_*, in denen Informationen zu korrelierten Ereignissen gespeichert sind.

Spaltenname	Datentyp	Kommentar
PARENT_EVT_ID	uniqueidentifier	Ereignis-UUID (Universal Unique Identifier) für übergeordnetes Ereignis
CHILD_EVT_ID	uniqueidentifier	Ereignis-UUID (Universal Unique Identifier) für untergeordnetes Ereignis
PARENT_EVT_TIME	DATETIME	Erstellungsdatum von übergeordnetem Ereignis
CHILD_EVT_TIME	DATETIME	Erstellungsdatum von untergeordnetem Ereignis
DATE_CREATED	DATE	Von DAS generiertes Einfügedatum
DATE_MODIFIED	DATETIME	Datum der letzten Aktualisierung
CREATED_BY	INT	Benutzer-ID beim Einfügen
MODIFIED_BY	INT	Benutzer-ID für letzte Aktualisierung

CORRELATED_EVENTS_RPT_V1

In dieser Ansicht sind aktuelle und korrelierte Verlaufsereignisse (korrelierte Ereignisse, die aus Archiven importiert wurden) enthalten.

Spaltenname	Datentyp	Kommentar
PARENT_EVT_ID	uniqueidentifier	Ereignis-UUID (Universal Unique Identifier) für übergeordnetes Ereignis
CHILD_EVT_ID	uniqueidentifier	Ereignis-UUID (Universal Unique Identifier) für untergeordnetes Ereignis
PARENT_EVT_TIME	DATETIME	Zeit von übergeordnetem Ereignis
CHILD_EVT_TIME	DATETIME	Zeit von untergeordnetem Ereignis
DATE_CREATED	DATETIME	Von DAS generiertes Einfügedatum
DATE_MODIFIED	DATETIME	Datum der letzten Aktualisierung
CREATED_BY	INT	Benutzer-ID beim Einfügen
MODIFIED_BY	INT	Benutzer-ID für letzte Aktualisierung

CRITICALITY_RPT_V

Diese Ansicht verweist auf die Tabelle CRIT_LKUP, in der Informationen zur Bestandsgefährlichkeit enthalten sind.

Spaltenname	Datentyp	Kommentar
CRITICALITY_CODE	varchar(5)	Code für Bestandsgefährlichkeit
CRITICALITY_NAME	varchar(50)	Name für Bestandsgefährlichkeit
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Nach Benutzer-ID
MODIFIED_BY	int	Nach Benutzer-ID

CUST_RPT_V

Diese Ansicht verweist auf die Tabelle CUST, in der Kundeninformationen für MSSPs gespeichert sind.

Spaltenname	Datentyp	Kommentar
CUSTOMER_ID	bigint	Kundenkennung
CUSTOMER_NAME	varchar(255)	Kundenname
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung

ENTITY_TYPE_RPT_V

Diese Ansicht verweist auf die Tabelle ENTITY_TYP, in der Informationen zu den Entitätstypen (Person, Organisation) gespeichert sind.

Spaltenname	Datentyp	Kommentar
ENTITY_TYPE_CODE	varchar(5)	Code für Entitätstyp
ENTITY_TYPE_NAME	varchar(50)	Name für Entitätstyp
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung

ENV_IDENTITY_RPT_V

Diese Ansicht verweist auf die Tabelle ENV_IDENTITY_LKUP, in der Informationen zur Umgebungsidentität für den Bestand gespeichert sind.

Spaltenname	Datentyp	Kommentar
ENVIRONMENT_IDENTITY_CODE	varchar(5)	Umgebungsidentitätscode
ENVIRONMENT_IDENTITY_NAME	varchar(255)	Umgebungsidentitätsname
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung

ESEC_DISPLAY_RPT_V

Diese Ansicht verweist auf die Tabelle ESEC_DISPLAY, in der die anzeigbaren Eigenschaften von Objekten gespeichert sind. Diese werden zurzeit beim Umbenennen von META-Tags verwendet. Sie werden mit der Ereigniskonfiguration (Unternehmensrelevanz) verwendet.

Spaltenname	Datentyp	Kommentar
DISPLAY_OBJECT	VARCHAR(32)	Das übergeordnete Objekt der Eigenschaft
TAG	VARCHAR(32)	Der native Tagname der Eigenschaft
LABEL	VARCHAR(32)	Die Anzeigezeichenkette des Tags.
POSITION	INT	Position des Tags in der Anzeige.
WIDTH	INT	Die Spaltenbreite
ALIGNMENT	INT	Die horizontale Ausrichtung
FORMAT	INT	Der einzeln aufgezählte Formatierer zum Anzeigen der Eigenschaft
ENABLED	BIT	Gibt an, ob das Tag angezeigt wird.
TYPE	INT	Gibt den Datentyp des Tags an. 1 = string 2 = ulong 3 = date 4 = uuid 5 = ipv4

Spaltenname	Datentyp	Kommentar
DESCRIPTION	VARCHAR(255)	Textbeschreibung des Tags
DATE_CREATED	DATETIME	Einfügedatum.
DATE_MODIFIED	DATETIME	Datum der letzten Aktualisierung.
CREATED_BY	INT	Benutzer-ID beim Einfügen.
MODIFIED_BY	INT	Benutzer-ID für letzte Aktualisierung.
REF_CONFIG	VARCHAR(4000)	Referenzielle Datenkonfiguration

ESEC_PORT_REFERENCE_RPT_V

Diese Ansicht verweist auf die Tabelle ESEC_PORT_REFERENCE, in der die gemäß Branchenstandard zugewiesenen Portnummern gespeichert sind.

Spaltenname	Datentyp	Kommentar
PORT_NUMBER	INT	Gemäß http://www.iana.org/assignments/port-numbers die numerische Darstellung des Ports. Diese Portnummer ist im Allgemeinen der Transportprotokollebene im TCP/IP-Stapel zugewiesen.
PROTOCOL_NUMBER	INT	Gemäß http://www.iana.org/assignments/protocol-numbers die numerischen Kennungen zum Darstellen von Protokollen, die in ein IP-Paket eingekapselt sind.
PORT_KEYWORD	VARCHAR(64)	Gemäß http://www.iana.org/assignments/port-numbers die Schlüsselwort-Darstellung des Ports.
PORT_DESCRIPTION	VARCHAR(512)	Portbeschreibung.
DATE_CREATED	DATETIME	Einfügedatum.
DATE_MODIFIED	DATETIME	Datum der letzten Aktualisierung
CREATED_BY	INT	Benutzer-ID beim Einfügen.
MODIFIED_BY	INT	Benutzer-ID für letzte Änderung.

ESEC_PROTOCOL_REFERENCE_RPT_V

Diese Ansicht verweist auf die Tabelle ESEC_PROTOCOL_REFERENCE, in der die gemäß Branchenstandard zugewiesenen Protokollnummern gespeichert sind.

Spaltenname	Datentyp	Kommentar
PROTOCOL_NUMBER	INT	Gemäß http://www.iana.org/assignments/protocol-numbers die numerischen Kennungen zum Darstellen von Protokollen, die in ein IP-Paket eingekapselt sind.

Spaltenname	Datentyp	Kommentar
PROTOCOL_KEYWORD	VARCHAR(64)	Gemäß http://www.iana.org/assignments/protocol-numbers das Schlüsselwort zum Darstellen von Protokollen, die in ein IP-Paket eingekapselt sind.
PROTOCOL_DESCRIPTION	VARCHAR(512)	IP-Paket-Protokollbeschreibung.
DATE_CREATED	DATETIME	Einfügedatum.
DATE_MODIFIED	DATETIME	Datum der letzten Aktualisierung.
CREATED_BY	INT	Benutzer-ID beim Einfügen.
MODIFIED_BY	INT	Benutzer-ID für letzte Aktualisierung.

ESEC_SEQUENCE _RPT_V

Diese Ansicht verweist auf die Tabelle ESEC_SEQUENCE, mit der Primärschlüssel-Sequenznummern für Sentinel-Tabellen generiert werden.

Spaltenname	Datentyp	Kommentar
TABLE_NAME	VARCHAR(32)	Name der Tabelle.
COLUMN_NAME	VARCHAR(32)	Name der Spalte
SEED	INT	Aktueller Wert des Primärschlüsselfelds.
DATE_CREATED	DATETIME	Einfügedatum.
DATE_MODIFIED	DATETIME	Datum der letzten Aktualisierung.
CREATED_BY	INT	Benutzer-ID beim Einfügen.
MODIFIED_BY	INT	Benutzer-ID für letzte Aktualisierung.

EVENTS_ALL_RPT_V (aus Gründen der Abwärtskompatibilität angegeben)

In dieser Ansicht sind aktuelle und Verlaufereignisse (Ereignisse, die aus Archiven importiert wurden) enthalten.

Spaltenname	Datentyp	Kommentar
EVENT_ID	uniqueidentifier	Ereigniskennung
RESOURCE_NAME	varchar(255)	Ressourcenname
SUB_RESOURCE	varchar(255)	Teilressourcenname
SEVERITY	int	Ereignisschweregrad
EVENT_PARSE_TIME	datetime	Ereigniszeit
EVENT_DATETIME	datetime	Ereigniszeit
BASE_MESSAGE	varchar(4000)	Basisnachricht
EVENT_NAME	varchar(255)	Der Name des Ereignisses, wie vom Sensor gemeldet
EVENT_TIME	varchar(255)	Zeit des Ereignisses, wie vom Sensor gemeldet
SENSOR_NAME	varchar(255)	Sensorname

Spaltenname	Datentyp	Kommentar
SENSOR_TYPE	varchar(5)	Sensortyp: H – Host-basiert N – Netzwerkbasier V – Virus O – Sonstiges
PROTOCOL	varchar(255)	Protokollname
SOURCE_IP	int	Quellen-IP-Adresse in numerischem Format
SOURCE_HOST_NAME	varchar(255)	Quell-Hostname
SOURCE_PORT	varchar(32)	Quell-Port
DESTINATION_IP	int	Ziel-IP-Adresse in numerischem Format
DESTINATION_HOST_NAME	varchar(255)	Ziel-Hostname
DESTINATION_PORT	varchar(32)	Ziel-Port
SOURCE_USER_NAME	varchar(255)	Quellenbenutzername
DESTINATION_USER_NAME	varchar(255)	Zielbenutzername
FILE NAME	varchar(1000)	Dateiname
EXTENDED_INFO	varchar(1000)	Erweiterte Informationen
REPORT_NAME	varchar(255)	Reportername
PRODUCT_NAME	varchar(255)	Berichtsproduktname
CUSTOM_TAG_1	varchar(255)	Kunden-Tag 1
CUSTOM_TAG_2	varchar(255)	Kunden-Tag 2
CUSTOM_TAG_3	int	Kunden-Tag 3
RESERVED_TAG_1	VARCHAR(255)	Reserviertes Tag 1 Für die zukünftige Verwendung durch Sentinel reserviert. Dieses Feld wird für Advisor-Informationen hinsichtlich Angriffsbeschreibungen verwendet.
RESERVED_TAG_2	varchar(255)	Für die zukünftige Verwendung durch Sentinel reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RESERVED_TAG_3	int	Für die zukünftige Verwendung durch Sentinel reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
SOURCE_UUID	uniqueidentifier	Quellen-UUID
PORT	varchar(64)	Collector-Port
AGENT	varchar(64)	Collector name
VULNERABILITY_RATING	int	Anfälligkeits-Rating
CRITICALITY_RATING	int	Gefährlichkeits-Rating

Spaltenname	Datentyp	Kommentar
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen.
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung.
RV01–10	INT	Reservierte Werte 1 bis 10 Für die zukünftige Verwendung durch Sentinel reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV11–20	DATETIME	Reservierte Werte 11 bis 20 Für die zukünftige Verwendung durch Sentinel reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV21–25	uniqueidentifier	Reservierte Werte 21 bis 25 Für die zukünftige Verwendung durch Sentinel zum Speichern von UUIDs reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV26–31	VARCHAR(255)	Reservierte Werte 26 bis 31 Für die zukünftige Verwendung durch Sentinel reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV32	VARCHAR(255)	Reservierter Wert 32 Reserviert für DeviceCategory Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.

Spaltenname	Datentyp	Kommentar
RV33	VARCHAR(255)	Reservierter Wert 33 Reserviert für EventContex Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV34	VARCHAR(255)	Reservierter Wert 34 Reserviert für SourceThreatLevel Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV35	VARCHAR(255)	Reservierter Wert 35 Reserviert für SourceUserContext. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV36	VARCHAR(255)	Reservierter Wert 36 Reserviert für DataContex. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV37	VARCHAR(255)	Reservierter Wert 37 Reserviert für SourceFunction. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV38	VARCHAR(255)	Reservierter Wert 38 Reserviert für SourceOperationalContext. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.

Spaltenname	Datentyp	Kommentar
RV39	VARCHAR(255)	Reservierter Wert 39 Reserviert für MSSPCustomerName. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV40–43	VARCHAR(255)	Reservierte Werte 40 bis 43 Für die zukünftige Verwendung durch Sentinel reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV44	VARCHAR(255)	Reservierter Wert 44 Reserviert für DestinationThreatLevel. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV45	VARCHAR(255)	Reservierter Wert 45 Reserviert für DestinationUserContext. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV46	VARCHAR(255)	Reservierter Wert 46 Reserviert für VirusStatus. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV47	VARCHAR(255)	Reservierter Wert 47 Für die zukünftige Verwendung durch Sentinel reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.

Spaltenname	Datentyp	Kommentar
RV48	VARCHAR(255)	Reservierter Wert 48 Reserviert für DestinationOperationalContext . Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV49	VARCHAR(255)	Reservierter Wert 49 Für die zukünftige Verwendung durch Sentinel reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV50	VARCHAR(255)	Taxonomiestufe 1
RV51	VARCHAR(255)	Taxonomiestufe 2
RV52	VARCHAR(255)	Taxonomiestufe 3
RV53	VARCHAR(255)	Taxonomiestufe 4
CV01–10	INT	Kundenwerte 1 bis 10 Reserviert für die Verwendung durch den Kunden, im Allgemeinen für die Zuordnung unternehmensrelevanter Daten
CV11–20	DATETIME	Kundenwerte 11 bis 20 Reserviert für die Verwendung durch den Kunden, im Allgemeinen für die Zuordnung unternehmensrelevanter Daten
CV21–100	VARCHAR(255)	Kundenwerte 21–100 Reserviert für die Verwendung durch den Kunden, im Allgemeinen für die Zuordnung unternehmensrelevanter Daten

EVENTS_ALL_RPT_V1 (aus Gründen der Abwärtskompatibilität angegeben)

In der Ansicht sind aktuelle Ereignisse enthalten. Es werden dieselben Spalten wie für EVENT_ALL_RPT_V verwendet.

EVENTS_RPT_V (aus Gründen der Abwärtskompatibilität angegeben)

In dieser Ansicht sind aktuelle und Verlaufereignisse enthalten. Es werden dieselben Spalten wie für EVENT_ALL_RPT_V verwendet.

EVENTS_RPT_V1 (aus Gründen der Abwärtskompatibilität angegeben)

In der Ansicht sind aktuelle Ereignisse enthalten. Es werden dieselben Spalten wie für EVENT_ALL_RPT_V verwendet.

EVENTS_RPT_V2 (aus Gründen der Abwärtskompatibilität angegeben)

In dieser Ansicht sind aktuelle Ereignisse und Verlaufsereignisse enthalten.

Spaltenname	Datentyp	Kommentar
EVENT_ID	uniqueidentifier	Ereigniskennung
RESOURCE_NAME	varchar(255)	Ressourcenname
SUB_RESOURCE	varchar(255)	Teilressourcenname
SEVERITY	int	Ereignisschweregrad
EVENT_PARSE_TIME	datetime	Ereigniszeit
EVENT_DATETIME	datetime	Ereigniszeit
BASE_MESSAGE	varchar(4000)	Basisnachricht
EVENT_NAME	varchar(255)	Der Name des Ereignisses, wie vom Sensor gemeldet
EVENT_TIME	varchar(255)	Zeit des Ereignisses, wie vom Sensor gemeldet
TAXONOMY_ID	bigint	Taxonomiekennung
PROTOCOL_ID	bigint	Protokollkennung
AGENT_ID	bigint	Collectorkennung
SOURCE_IP	int	Quellen-IP-Adresse in numerischem Format
SOURCE_HOST_NAME	varchar(255)	Quell-Hostname
SOURCE_PORT	varchar(32)	Quell-Port
DESTINATION_IP	int	Ziel-IP-Adresse in numerischem Format
DESTINATION_HOST_NAME	varchar(255)	Ziel-Hostname
DESTINATION_PORT	varchar(32)	Ziel-Port
SOURCE_USER_NAME	varchar(255)	Quellenbenutzername
DESTINATION_USER_NAME	varchar(255)	Zielbenutzername
FILE NAME	varchar(1000)	Dateiname
EXTENDED_INFO	varchar(1000)	Erweiterte Informationen
CUSTOM_TAG_1	varchar(255)	Kunden-Tag 1
CUSTOM_TAG_2	varchar(255)	Kunden-Tag 2
CUSTOM_TAG_3	int	Kunden-Tag 3
RESERVED_TAG_1	VARCHAR(255)	Reserviertes Tag 1 Für die zukünftige Verwendung durch Sentinel reserviert. Dieses Feld wird für Advisor-Informationen hinsichtlich Angriffsbeschreibungen verwendet.

Spaltenname	Datentyp	Kommentar
RESERVED_TAG_2	varchar(255)	Für die zukünftige Verwendung durch Sentinel reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RESERVED_TAG_3	int	Für die zukünftige Verwendung durch Sentinel reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
VULNERABILITY_RATING	int	Anfälligkeits-Rating
CRITICALITY_RATING	int	Gefährlichkeits-Rating
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen.
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung.
RV01–10	INT	Reservierte Werte 1 bis 10 Für die zukünftige Verwendung durch Sentinel reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV11–20	DATETIME	Reservierte Werte 1 bis 31 Für die zukünftige Verwendung durch Sentinel reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV21–25	uniqueidentifier	Reservierte Werte 21 bis 25 Für die zukünftige Verwendung durch Sentinel zum Speichern von UUIDs reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.

Spaltenname	Datentyp	Kommentar
RV26–31	VARCHAR(255)	Reservierte Werte 26 bis 31 Für die zukünftige Verwendung durch Sentinel reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV33	VARCHAR(255)	Reservierter Wert 33 Reserviert für EventContext Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV34	VARCHAR(255)	Reservierter Wert 34 Reserviert für SourceThreatLevel Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV35	VARCHAR(255)	Reservierter Wert 35 Reserviert für SourceUserContext. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV36	VARCHAR(255)	Reservierter Wert 36 Reserviert für DataContext. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV37	VARCHAR(255)	Reservierter Wert 37 Reserviert für SourceFunction. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.

Spaltenname	Datentyp	Kommentar
RV38	VARCHAR(255)	Reservierter Wert 38 Reserviert für SourceOperationalContext. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV40–43	VARCHAR(255)	Reservierte Werte 40 bis 43 Für die zukünftige Verwendung durch Sentinel reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV44	VARCHAR(255)	Reservierter Wert 44 Reserviert für DestinationThreatLevel. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV45	VARCHAR(255)	Reservierter Wert 45 Reserviert für DestinationUserContext. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV46	VARCHAR(255)	Reservierter Wert 46 Reserviert für VirusStatus. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV47	VARCHAR(255)	Reservierter Wert 47 Für die zukünftige Verwendung durch Sentinel reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.

Spaltenname	Datentyp	Kommentar
RV48	VARCHAR(255)	Reservierter Wert 48 Reserviert für DestinationOperationalContext. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
RV49	VARCHAR(255)	Reservierter Wert 49 Für die zukünftige Verwendung durch Sentinel reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
REFERENCE_ID 01–20	BIGINT	Für die zukünftige Verwendung durch Sentinel reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
CV01–10	INT	Kundenwerte 1 bis 10 Reserviert für die Verwendung durch den Kunden, im Allgemeinen für die Zuordnung unternehmensrelevanter Daten
CV11–20	DATETIME	Kundenwerte 11 bis 20 Reserviert für die Verwendung durch den Kunden, im Allgemeinen für die Zuordnung unternehmensrelevanter Daten
CV21–100	VARCHAR(255)	Kundenwerte 21–100 Reserviert für die Verwendung durch den Kunden, im Allgemeinen für die Zuordnung unternehmensrelevanter Daten

EVT_AGENT_RPT_V

Diese Ansicht verweist auf die Tabelle EVT_AGENT, in der Informationen zu den Collectors gespeichert sind.

Spaltenname	Datentyp	Kommentar
AGENT_ID	bigint	Collectorkennung
AGENT	varchar(64)	Collector name
PORT	varchar(64)	Collector-Port
REPORT_NAME	varchar(255)	Reportername
PRODUCT_NAME	varchar(255)	Produktname
SENSOR_NAME	varchar(255)	Sensorname

Spaltenname	Datentyp	Kommentar
SENSOR_TYPE	varchar(5)	Sensortyp: H – Host-basiert N – Netzwerkbasiert V – Virus O – Sonstiges
DEVICE_CTGRY	varchar(255)	Gerätekategorie
SOURCE_UUID	uniqueidentifier	UUID (Universal Unique Identifier) der Quellkomponente
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen.
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung.

EVT_ASSET_RPT_V

Diese Ansicht verweist auf die Tabelle EVT_ASSET, in der Bestandsinformationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
EVENT_ASSET_ID	bigint	Ereignisbestandskennung
ASSET_NAME	varchar(255)	Bestandsname
PHYSICAL_ASSET_NAME	varchar(255)	Name für physischen Bestand
REFERENCE_ASSET_ID	varchar(100)	Referenzbestandskennung, stellt Verknüpfung zu Quellbestands-Verwaltungssystem bereit.
MAC_ADDRESS	varchar(100)	MAC-Adresse
RACK_NUMBER	varchar(50)	Regalnummer
ROOM_NAME	varchar(100)	Raumname
BUILDING_NAME	varchar(255)	Gebäudenname
CITY	varchar(100)	Ort
STATE	varchar(100)	Status
COUNTRY	varchar(100)	Land
ZIP_CODE	varchar(50)	Postleitzahl
ASSET_CATEGORY_NAME	varchar(100)	Bestandskategorienname
NETWORK_IDENTITY_NAME	varchar(255)	Netzwerkidentitätsname für Bestand
ENVIRONMENT_IDENTITY_NAME	varchar(255)	Umgebungsname
ASSET_VALUE_NAME	varchar(50)	Bestandswertname
CRITICALITY_NAME	varchar(50)	Name für Bestandsgefährlichkeit
SENSITIVITY_NAME	varchar(50)	Bestandsvertraulichkeitsname
CONTACT_NAME_1	varchar(255)	Name der Kontaktperson/-organisation 1
CONTACT_NAME_2	varchar(255)	Name der Kontaktperson/-organisation 2
ORGANIZATION_NAME_1	varchar(100)	Bestandseigentümer-Organisationsebene 1
ORGANIZATION_NAME_2	varchar(100)	Bestandseigentümer-Organisationsebene 2

Spaltenname	Datentyp	Kommentar
ORGANIZATION_NAME_3	varchar(100)	Bestandseigentümer-Organisationsebene 3
ORGANIZATION_NAME_4	varchar(100)	Bestandseigentümer-Organisationsebene 4
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen.
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung.

EVT_DEST_EVT_NAME_SMRY_1_RPT_V

In dieser Ansicht wird die Anzahl der Ereignisse nach Ziel, Taxonomie, Ereignisname, Schweregrad und Ereigniszeit zusammengefasst.

Spaltenname	Datentyp	Kommentar
DESTINATION_IP	int	Ziel-IP-Adresse
DESTINATION_EVENT_ASSET_ID	bigint	Ereignisbestandskennung
TAXONOMY_ID	bigint	Taxonomiekennung
EVENT_NAME_ID	bigint	Ereignisnamenkennung
SEVERITY	int	Ereignisschweregrad
CUSTOMER_ID	bigint	Kundenkennung
EVT_TIME	datetime	Ereigniszeit
EVT_COUNT	int	Anzahl Ereignisse
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen.
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung.

EVT_DEST_SMRY_1_RPT_V

In dieser Ansicht sind zusammenfassende Informationen zum Ereignisziel enthalten.

Spaltenname	Datentyp	Kommentar
DESTINATION_IP	int	Ziel-IP-Adresse
DESTINATION_EVENT_ASSET_ID	bigint	Ereignisbestandskennung
DESTINATION_PORT	varchar(32)	Ziel-Port
DESTINATION_USR_ID	bigint	Zielbenutzerkennung
TAXONOMY_ID	bigint	Taxonomiekennung
EVENT_NAME_ID	bigint	Ereignisnamenkennung
RESOURCE_ID	bigint	Ressourcenkennung
AGENT_ID	bigint	Collectorkennung
PROTOCOL_ID	bigint	Protokollkennung
SEVERITY	int	Ereignisschweregrad
CUSTOMER_ID	bigint	Kundenkennung
EVENT_TIME	datetime	Ereigniszeit
EVENT_COUNT	int	Anzahl Ereignisse
DATE_CREATED	datetime	Einfügedatum

Spaltenname	Datentyp	Kommentar
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen.
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung.

EVT_DEST_TXNMY_SMRY_1_RPT_V

In dieser Ansicht wird die Anzahl der Ereignisse nach Ziel, Taxonomie, Schweregrad und Ereigniszeit zusammengefasst.

Spaltenname	Datentyp	Kommentar
DESTINATION_IP	int	Ziel-IP-Adresse
DESTINATION_EVENT_ASSET_ID	bigint	Ereignisbestandskennung
TAXONOMY_ID	bigint	Taxonomiekennung
SEVERITY	int	Ereignisschweregrad
CUSTOMER_ID	bigint	Kundenkennung
EVENT_TIME	datetime	Ereigniszeit
EVENT_COUNT	int	Anzahl Ereignisse
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen.
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung.

EVT_NAME_RPT_V

Diese Ansicht verweist auf die Tabelle EVT_NAME, in der Informationen zu Ereignisnamen gespeichert sind.

Spaltenname	Datentyp	Kommentar
EVENT_NAME_ID	bigint	Ereignisnamenkennung
EVENT_NAME	varchar(255)	Ereignisname
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen.
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung.

EVT_PORT_SMRY_1_RPT_V

In dieser Ansicht wird die Anzahl der Ereignisse nach Ziel-Port, Schweregrad und Ereigniszeit zusammengefasst.

Spaltenname	Datentyp	Kommentar
DESTINATION_PORT	Varchar(32)	Ziel-Port
SEVERITY	int	Ereignisschweregrad
CUSTOMER_ID	bigint	Kundenkennung
EVENT_TIME	datetime	Ereigniszeit
EVENT_COUNT	int	Anzahl Ereignisse
DATE_CREATED	datetime	Einfügedatum

Spaltenname	Datentyp	Kommentar
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen.
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung.

EVT_PRTCL_RPT_V

Diese Ansicht verweist auf die Tabelle EVT_PRTCL, in der die Ereignisprotokoll-Informationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
PROTOCOL_ID	bigint	Protokollkennung
PROTOCOL_NAME	varchar(255)	Protokollname
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen.
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung.

EVT_RSRC_RPT_V

Diese Ansicht verweist auf die Tabelle EVT_RSRC, in der die Ereignisressourcen-Informationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
RESOURCE_ID	bigint	Ressourcenkennung
RESOURCE_NAME	varchar(255)	Ressourcenname
SUB_RESOURCE_NAME	varchar(255)	Teilressourcenname
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen.
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung.

EVT_SEV_SMRY_1_RPT_V

In dieser Ansicht wird die Anzahl der Ereignisse nach Schweregrad und Ereigniszeit zusammengefasst.

Spaltenname	Datentyp	Kommentar
SEVERITY	int	Ereignisschweregrad
CUSTOMER_ID	bigint	Kundenkennung
EVENT_TIME	datetime	Ereigniszeit
EVENT_COUNT	int	Anzahl Ereignisse
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen.
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung.

EVT_SRC_SMRY_1_RPT_V

In dieser Ansicht sind zusammenfassende Informationen zu Ereignisquelle und -ziel enthalten.

Spaltenname	Datentyp	Kommentar
SOURCE_IP	int	Quellen-IP-Adresse
SOURCE_EVENT_ASSET_ID	bigint	Ereignisbestandskennung
SOURCE_PORT	varchar(32)	Quell-Port
SOURCE_USER_ID	bigint	Benutzerkennung
TAXONOMY_ID	bigint	Taxonomiekennung
EVENT_NAME_ID	bigint	Ereignisnamenkennung
RESOURCE_ID	bigint	Ressourcenkennung
AGENT_ID	bigint	Collectorkennung
PROTOCOL_ID	bigint	Protokollkennung
SEVERITY	int	Ereignisschweregrad
CUSTOMER_ID	bigint	Kundenkennung
EVENT_TIME	datetime	Ereigniszeit
EVENT_COUNT	int	Anzahl Ereignisse
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen.
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung.

EVT_TXNMY_RPT_V

Diese Ansicht verweist auf die Tabelle EVT_TXNMY, in der die Ereignistaxonomie-Informationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
TAXONOMY_ID	bigint	Taxonomiekennung
TAXONOMY_LEVEL_1	varchar(100)	Taxonomiestufe 1
TAXONOMY_LEVEL_2	varchar(100)	Taxonomiestufe 2
TAXONOMY_LEVEL_3	varchar(100)	Taxonomiestufe 3
TAXONOMY_LEVEL_4	varchar(100)	Taxonomiestufe 4
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen.
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung.
TAXONOMY_ID	bigint	Taxonomiekennung

EVT_USR_RPT_V

Diese Ansicht verweist auf die Tabelle EVT_USR, in der Benutzerinformationen zu Ereignissen gespeichert sind.

Spaltenname	Datentyp	Kommentar
USER_ID	bigint	Benutzerkennung
USER_NAME	varchar(255)	Benutzername
DATE_CREATED	datetime	Einfügedatum

Spaltenname	Datentyp	Kommentar
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen.
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung.
USER_ID	bigint	Benutzerkennung

EXTERNAL_DATA_RPT_V

Diese Ansicht verweist auf die Tabelle EXTERNAL_DATA, in der externe Daten gespeichert sind.

Spaltenname	Datentyp	Kommentar
EXTERNAL_DATA_ID	int	Kennung für externe Daten
SOURCE_NAME	varchar(50)	Quellname
SOURCE_DATA_ID	varchar(255)	Quelldatenkennung
EXTERNAL_DATA	text	Externe Daten
EXTERNAL_DATA_TYPE	varchar(10)	Externer Datentyp
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen.
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung.

HIST_EVENTS_RPT_V

In dieser Ansicht sind Verlaufsereignisse (Ereignisse, die aus Archiven wiederhergestellt wurden) enthalten.

HIST_INCIDENTS_RPT_V

In dieser Ansicht sind Laufsvorfälle (Vorfälle, die aus Archiven wiederhergestellt wurden) enthalten.

IMAGES_RPT_V

Diese Ansicht verweist auf die Tabelle IMAGES, in der Bildinformationen zur Systemübersicht gespeichert sind.

Spaltenname	Datentyp	Kommentar
NAME	VARCHAR(128)	Bildname
TYPE	VARCHAR(64)	Bildtyp
DATA	TEXT	Bilddaten
DATE_CREATED	DATETIME	Einfügedatum
DATE_MODIFIED	DATETIME	Datum der letzten Aktualisierung
CREATED_BY	INT	Benutzer-ID beim Einfügen
MODIFIED_BY	INT	Benutzer-ID für letzte Aktualisierung

INCIDENTS_ASSETS_RPT_V

Diese Ansicht verweist auf die Tabelle INCIDENTS_ASSETS, in der Informationen zu den Beständen gespeichert sind, die die Vorfälle darstellen, die in der Sentinel Console erstellt werden.

Spaltenname	Datentyp	Kommentar
INC_ID	INT	Vorfallkennung – Sequenznummer
ASSET_ID	uniqueidentifier	Bestands-UUID (Universal Unique Identifier)
DATE_CREATED	DATETIME	Einfügedatum
DATE_MODIFIED	DATETIME	Datum der letzten Aktualisierung
CREATED_BY	INT	Benutzer-ID beim Einfügen
MODIFIED_BY	INT	Benutzer-ID für letzte Aktualisierung

INCIDENTS_EVENTS_RPT_V

Diese Ansicht verweist auf die Tabelle INCIDENTS_EVENTS, in der Informationen zu den Ereignissen gespeichert sind, die die Vorfälle darstellen, die in der Sentinel Console erstellt werden.

Spaltenname	Datentyp	Kommentar
INC_ID	INT	Vorfallkennung – Sequenznummer
EVT_ID	uniqueidentifier	Ereignis-UUID (Universal Unique Identifier)
EVT_TIME	DATETIME	Ereigniszeit
DATE_CREATED	DATETIME	Einfügedatum
DATE_MODIFIED	DATETIME	Datum der letzten Aktualisierung
CREATED_BY	INT	Benutzer-ID beim Einfügen
MODIFIED_BY	INT	Benutzer-ID für letzte Aktualisierung

INCIDENTS_RPT_V

Diese Ansicht verweist auf die Tabelle INCIDENTS, in der Informationen gespeichert sind, die die Details zu den Vorfällen beschreiben, die in der Sentinel Console erstellt werden.

Spaltenname	Datentyp	Kommentar
INC_ID	INT	Vorfallkennung – Sequenznummer
NAME	VARCHAR(255)	Vorfallname
SEVERITY	INT	Vorfallschweregrad
STT_ID	INT	Zustands-ID für Vorfall
SEVERITY_RATING	VARCHAR(32)	Durchschnitt der Schweregrade von Ereignissen, die den Vorfall bilden.
VULNERABILITY_RATING	VARCHAR(32)	Für die zukünftige Verwendung durch Sentinel reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.

Spaltenname	Datentyp	Kommentar
CRITICALITY_RATING	VARCHAR(32)	Für die zukünftige Verwendung durch Sentinel reserviert. Die Verwendung dieses Felds für andere Angaben kann dazu führen, dass Daten von zukünftigen Funktionen überschrieben werden.
DATE_CREATED	DATETIME	Einfügedatum
DATE_MODIFIED	DATETIME	Datum der letzten Aktualisierung
CREATED_BY	INT	Benutzer-ID beim Einfügen
MODIFIED_BY	INT	Benutzer-ID für letzte Aktualisierung
INC_DESC	varchar(4000)	Vorfallbeschreibung
INC_PRIORITY	int	Vorfallpriorität
INC_CAT	varchar(255)	Vorfallkategorie
INC_RES	varchar(4000)	Vorfallauflösung

INCIDENTS_VULN_RPT_V

Diese Ansicht verweist auf die Tabelle INCIDENTS_VULN, in der Informationen zu den Anfälligkeiten gespeichert sind, die die Vorfälle darstellen, die in der Sentinel Console erstellt werden.

Spaltenname	Datentyp	Kommentar
INC_ID	INT	Vorfallkennung – Sequenznummer
VULN_ID	uniqueidentifier	Anfälligkeits-UUID (Universal Unique Identifier)
DATE_CREATED	DATETIME	Einfügedatum
DATE_MODIFIED	DATETIME	Datum der letzten Aktualisierung
CREATED_BY	INT	Benutzer-ID beim Einfügen
MODIFIED_BY	INT	Benutzer-ID für letzte Aktualisierung

L_STAT_RPT_V

Diese Ansicht verweist auf die Tabelle L_STAT, in der statistische Informationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
RES_NAME	VARCHAR(32)	Ressourcenname
STATS_NAME	VARCHAR(32)	Name der Statistik
STATS_VALUE	VARCHAR(32)	Wert der Statistik
OPEN_TOT_SECS	NUMERIC	Anzahl von Sekunden seit 1970.

LOGS_RPT_V

Diese Ansicht verweist auf die Tabelle LOGS_RPT, in der Protokollinformationen gespeichert sind.

PROTOKOLL-Tabelle		
Spaltenname	Datentyp	Kommentar
LOG_ID	NUMBER	Sequenznummer
TIME	DATE	Datum des Protokolls
MODULE	VARCHAR(64)	Modulprotokoll ist für
TEXT	VARCHAR(4000)	Protokolltext

NETWORK_IDENTITY_RPT_V

Diese Ansicht verweist auf die Tabelle NETWORK_IDENTITY_LKUP, in der Netzwerkidentitätsinformationen für den Bestand gespeichert sind.

Spaltenname	Datentyp	Kommentar
NETWORK_IDENTITY_CD	varchar(5)	Netzwerkidentitätscode
NETWORK_IDENTITY_NAME	varchar(255)	Netzwerkidentitätsname
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung

ORGANIZATION_RPT_V

Diese Ansicht verweist auf die Tabelle ORGANIZATION, in der Informationen zur Organisation (zum Bestand) gespeichert sind.

Spaltenname	Datentyp	Kommentar
ORGANIZATION_ID	uniqueidentifier	Organisationskennung
ORGANIZATION_NAME	varchar(100)	Name der Organisation
CUSTOMER_ID	bigint	Kundenkennung
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung

PERSON_RPT_V

Diese Ansicht verweist auf die Tabelle PERSON, in der persönliche Informationen (Informationen zum Bestand) gespeichert sind.

Spaltenname	Datentyp	Kommentar
PERSON_ID	uniqueidentifier	Personenkennung
FIRST_NAME	varchar(255)	Vorname
LAST_NAME	varchar(255)	Nachname
CUSTOMER_ID	bigint	Kundenkennung
PHONE_NUMBER	varchar(50)	Telefonnummer
EMAIL_ADDRESS	varchar(255)	Email-Adresse
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung

Spaltenname	Datentyp	Kommentar
CREATED_BY	int	Benutzer-ID beim Einfügen
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung

PHYSICAL_ASSET_RPT_V

Diese Ansicht verweist auf die Tabelle PHYSICAL_ASSET, in der Informationen zum physischen Bestand gespeichert sind.

Spaltenname	Datentyp	Kommentar
PHYSICAL_ASSET_ID	uniqueidentifier	Kennung für physischen Bestand
CUSTOMER_ID	int	Kundenkennung
LOCATION_ID	bigint	Standortkennung
HOST_NAME	varchar(255)	Hostname
IP_ADDRESS	int	IP-Adresse
NETWORK_IDENTITY_CD	varchar(5)	Netzwerkidentitätscode
MAC_ADDRESS	varchar(100)	MAC-Adresse
RACK_NUMBER	varchar(50)	Regalnummer
ROOM_NAME	varchar(100)	Raumname
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung

PRODUCT_RPT_V

Diese Ansicht verweist auf die Tabelle PRDT, in der Bestands-Produktinformationen gespeichert sind.

Spaltenname	Datentyp	Kommentar
PRODUCT_ID	bigint	Produktkennung
PRODUCT_NAME	varchar(255)	Produktname
PRODUCT_VERSION	varchar(100)	Produktversion
VENDOR_ID	bigint	Händlerkennung
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung

ROLE_RPT_V

Diese Ansicht verweist auf die Tabelle ROLE_LKUP, in der Informationen zur Benutzerfunktion (Bestandsinformationen) gespeichert sind.

Spaltenname	Datentyp	Kommentar
ROLE_CODE	varchar(5)	Funktionscode
ROLE_NAME	varchar(255)	Funktionsname
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung

SENSITIVITY_RPT_V

Diese Ansicht verweist auf die Tabelle SENSITIVITY_LKUP, in der Informationen zur Bestandsvertraulichkeit gespeichert sind.

Spaltenname	Datentyp	Kommentar
SENSITIVITY_CODE	varchar(5)	Code für Bestandsvertraulichkeit
SENSITIVITY_NAME	varchar(50)	Bestandsvertraulichkeitsname
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Nach Benutzer-ID
MODIFIED_BY	int	Nach Benutzer-ID

STATES_RPT_V

Diese Ansicht verweist auf die Tabelle STATES, in der Definitionen zu den Zuständen gespeichert sind, die von Anwendungen oder vom Kontext definiert werden.

Spaltenname	Datentyp	Kommentar
STT_ID	INT	Zustands-ID – Sequenznummer
CONTEXT	VARCHAR(64)	Kontext zum Zustand. Zum Beispiel Fall, Vorfall, Benutzer.
NAME	VARCHAR(64)	Zustandsname.
TERMINAL_FLAG	VARCHAR(1)	Gibt an, ob der Zustand des Vorfalls aufgelöst ist.
DATE_CREATED	DATETIME	Einfügedatum
DATE_MODIFIED	DATETIME	Datum der letzten Aktualisierung
MODIFIED_BY	INT	Benutzer-ID beim Einfügen
CREATED_BY	INT	Benutzer-ID für letzte Aktualisierung

Ansicht UNASSIGNED_INCIDENTS_RPT_V

Diese Ansicht verweist auf die Tabellen CASES und INCIDENTS, um nicht zugeordnete Fälle zu melden.

Name	Datentyp
INC_ID	INT
NAME	VARCHAR(255)
SEVERITY	INT
STT_ID	INT
SEVERITY_RATING	VARCHAR(32)
VULNERABILITY_RATING	VARCHAR(32)
CRITICALITY_RATING	VARCHAR(32)
DATE_CREATED	DATETIME
DATE_MODIFIED	DATETIME
CREATED_BY	INT
MODIFIED_BY	INT
INC_DESC	VARCHAR(4000)
INC_PRIORITY	INT
INC_CAT	VARCHAR(255)
INC_RES	VARCHAR(4000)

USERS_RPT_V

Diese Ansicht verweist auf die Tabelle USERS, in der alle Benutzer der Anwendung gespeichert sind. Die Benutzer werden zur Unterstützung für Drittanbieter-Berichterstellungswerkzeuge auch als Datenbankbenutzer erstellt.

Spaltenname	Datentyp	Kommentar
USR_ID	INT	Benutzerkennung – Sequenznummer
NAME	VARCHAR(64)	Kurzer, eindeutiger Benutzername, der als Anmeldename verwendet wird
CNT_ID	INT	Kontakt-ID – Sequenznummer
STT_ID	INT	Zustands-ID. Status ist entweder aktiv oder inaktiv.
DESCRIPTION	VARCHAR(512)	Kommentare
DATE_CREATED	DATETIME	Einfügedatum
DATE_MODIFIED	DATETIME	Datum der letzten Aktualisierung
CREATED_BY	INT	Benutzer-ID beim Einfügen
MODIFIED_BY	INT	Benutzer-ID für letzte Aktualisierung
PERMISSIONS	VARCHAR(4000)	Dem Sentinel-Benutzer aktuell zugewiesene Benutzerberechtigung
FILTER	VARCHAR(128)	Dem Sentinel-Benutzer aktuell zugewiesener Sicherheitsfilter
UPPER_NAME	VARCHAR(64)	Benutzername in Großschreibung
DOMAIN_AUTH_IND	Bit	Domänenauthentifizierungskennung

VENDOR_RPT_V

Diese Ansicht verweist auf die Tabelle VNDR, in der Informationen zu Bestandsprodukt-Händlern gespeichert sind.

Spaltenname	Datentyp	Kommentar
VENDOR_ID	bigint	Händlerkennung
VENDOR_NAME	varchar(255)	Händlername
DATE_CREATED	datetime	Einfügedatum
DATE_MODIFIED	datetime	Datum der letzten Aktualisierung
CREATED_BY	int	Benutzer-ID beim Einfügen
MODIFIED_BY	int	Benutzer-ID für letzte Aktualisierung

VULN_CALC_SEVERITY_RPT_V

Diese Ansicht verweist auf die Tabellen VULN_RSRC und VULN zum Ermitteln des Schweregrad-Ratings für die eSecurity-Anfälligkeit auf der Grundlage aktueller Anfälligkeiten.

Spaltenname	Datentyp
RSRC_ID	uniqueidentifier
IP	VARCHAR(32)
HOST_NAME	VARCHAR(255)
CRITICALITY	int
ASSIGNED_VULN_SEVERITY	int
VULN_COUNT	Anzahl der Anfälligkeiten für angegebene Ressource

Spaltenname	Datentyp
CALC_SEVERITY	Ermittelter Schweregrad auf Grundlage von ASSIGNED_VULN_SEVERITY und CRITICALITY

VULN_CODE_RPT_V

Diese Ansicht verweist auf die Tabelle VULN_CODE, in der von der Branche zugeordnete Anfälligkeitscodes wie CVEs und CANs von Mitre gespeichert sind.

Spaltenname	Datentyp
VULN_CODE_ID	VARCHAR(36)
VULN_ID	VARCHAR(36)
VULN_CODE_TYPE	VARCHAR(64)
VULN_CODE_VALUE	VARCHAR(255)
URL	VARCHAR(512)
DATE_CREATED	DATETIME
DATE_MODIFIED	DATETIME
CREATED_BY	INT
MODIFIED_BY	INT

VULN_INFO_RPT_V

Diese Ansicht verweist auf die Tabelle VULN_INFO, in der zusätzliche Informationen gespeichert sind, die bei einem Absuchvorgang gemeldet wurden.

Spaltenname	Datentyp
VULN_INFO_ID	VARCHAR(36)
VULN_ID	VARCHAR(36)
VULN_INFO_TYPE	VARCHAR(36)
VULN_INFO_VALUE	VARCHAR(2000)
DATE_CREATED	DATETIME
DATE_MODIFIED	DATETIME
CREATED_BY	INT
MODIFIED_BY	INT

VULN_RPT_V

Diese Ansicht verweist auf die Tabelle VULN, in der Informationen zum abgesuchten System gespeichert sind. Jedes Absuchprogramm verfügt über einen eigenen Eintrag für jedes System.

Spaltenname	Datentyp
VULN_ID	VARCHAR(36)
RSRC_ID	VARCHAR(36)
PORT_NAME	VARCHAR(64)
PORT_NUMBER	INT
NETWORK_PROTOCOL	INT
APPLICATION_PROTOCOL	VARCHAR(64)
ASSIGNED_VULN_SEVERITY	INT
COMPUTED_VULN_SEVERITY	INT
VULN_DESCRIPTION	CLOB

Spaltenname	Datentyp
VULN_SOLUTION	CLOB
VULN_SUMMARY	VARCHAR(1000)
BEGIN_EFFECTIVE_DATE	DATETIME
END_EFFECTIVE_DATE	DATETIME
DETECTED_OS	VARCHAR(64)
DETECTED_OS_VERSION	VARCHAR(64)
SCANNED_APP	VARCHAR(64)
SCANNED_APP_VERSION	VARCHAR(64)
VULN_USER_NAME	VARCHAR(64)
VULN_USER_DOMAIN	VARCHAR(64)
VULN_TAXONOMY	VARCHAR(1000)
SCANNER_CLASSIFICATION	VARCHAR(255)
VULN_NAME	VARCHAR(300)
VULN_MODULE	VARCHAR(64)
DATE_CREATED	DATETIME
DATE_MODIFIED	DATETIME
CREATED_BY	INT
MODIFIED_BY	INT

VULN_RSRC_RPT_V

Diese Ansicht verweist auf die Tabelle VULN_RSRC, in der die einzelnen Ressourcen gespeichert sind, die für einen bestimmten Absuchvorgang abgesucht wurden.

Spaltenname	Datentyp
RSRC_ID	VARCHAR(36)
SCANNER_ID	VARCHAR(36)
IP	VARCHAR(32)
HOST_NAME	VARCHAR(255)
LOCATION	VARCHAR(128)
DEPARTMENT	VARCHAR(128)
BUSINESS_SYSTEM	VARCHAR(128)
OPERATIONAL_ENVIRONMENT	VARCHAR(64)
CRITICALITY	INT
REGULATION	VARCHAR(128)
REGULATION_RATING	VARCHAR(64)
DATE_CREATED	DATETIME
DATE_MODIFIED	DATETIME
CREATED_BY	INT
MODIFIED_BY	INT

VULN_RSRC_SCAN_RPT_V

Diese Ansicht verweist auf die Tabelle VULN_RSRC_SCAN, in der die einzelnen Ressourcen gespeichert sind, die für einen bestimmten Absuchvorgang abgesucht wurden.

Spaltenname	Datentyp
RSRC_ID	VARCHAR(36)
SCAN_ID	VARCHAR(36)
DATE_CREATED	DATETIME
DATE_MODIFIED	DATETIME
CREATED_BY	INT
MODIFIED_BY	INT

VULN_SCAN_RPT_V

Diese Ansicht verweist auf die Tabelle, in der Informationen zu Absuchvorgängen gespeichert sind.

Spaltenname	Datentyp
SCAN_ID	VARCHAR(36)
SCANNER_ID	VARCHAR(36)
SCAN_TYPE	VARCHAR(10)
SCAN_START_DATE	DATETIME
SCAN_END_DATE	DATETIME
CONSOLIDATION_SERVER	VARCHAR(64)
DATE_CREATED	DATETIME
DATE_MODIFIED	DATETIME
CREATED_BY	INT
MODIFIED_BY	INT

VULN_SCAN_VULN_RPT_V

Diese Ansicht verweist auf die Tabelle VULN_SCAN_VULN, in der die Anfälligkeiten gespeichert sind, die bei Absuchvorgängen ermittelt wurden.

Spaltenname	Datentyp
SCAN_ID	VARCHAR(36)
VULN_ID	VARCHAR(36)
DATE_CREATED	DATETIME
DATE_MODIFIED	DATETIME
CREATED_BY	INT
MODIFIED_BY	INT

VULN_SCANNER_RPT_V

Diese Ansicht verweist auf die Tabelle VULN_SCANNER, in der Informationen zu den Anfälligkeits-Absuchprogrammen gespeichert sind.

Spaltenname	Datentyp
SCANNER_ID	VARCHAR(36)
PRODUCT_NAME	VARCHAR(100)
PRODUCT_VERSION	VARCHAR(64)
SCANNER_TYPE	VARCHAR(64)
VENDOR	VARCHAR(100)
SCANNER_INSTANCE	VARCHAR(64)
DATE_CREATED	DATETIME
DATE_MODIFIED	DATETIME
CREATED_BY	INT
MODIFIED_BY	INT

A

Checkliste für die Sentinel-Fehlersuche

HINWEIS: Die Begriffe „Agent“ und „Collector“ sind austauschbar. Im Folgenden werden Agenten als „Collectors“ bezeichnet.

Diese Checkliste soll Sie bei der Problemdiagnose unterstützen. Durch das Ausfüllen dieser Checkliste kann der Großteil der gängigen Probleme mit geringerem Zeitaufwand gelöst werden. Für die Probleme, deren Lösung mehr Zeit in Anspruch nimmt, sind die Diagnoseinformationen bereits vorhanden; auf diese Weise wird unnötigem Diagnoseaufwand vorgebeugt.

Eintrag in Checkliste	Informationen	Beispiel
Novell-Version:		v5.1.3
Novell-Plattform und Betriebssystemversion:		Win2003 Server SP1
Datenbankplattform und Betriebssystemversion:		MS SQL 2000 SP3a
Sentinel Server-Hardwarekonfiguration <ul style="list-style-type: none"> ▪ Prozessor ▪ Arbeitsspeicher ▪ Sonstiges 		5 GB RAM 4 CPU 3,0 GHz
Datenbankserver-Hardwarekonfiguration <ul style="list-style-type: none"> ▪ Prozessor ▪ Arbeitsspeicher ▪ Sonstiges (falls separate Box) 		8 GB RAM 4 CPU 3,0 GHz
Datenbankspeicherkonfiguration (NAS, SAN, Lokal usw.)		Lokal mit Offsite-Sicherung
Betriebssystem und Konfiguration des Berichtsservers (Crystal Server)		Crystal XI Win2003 Server SP1 Windows-Authentifizierung

HINWEIS: Je nach Konfiguration (Verteilung) Ihres Sentinel-Systems muss die obige Tabelle möglicherweise erweitert werden. So sind beispielsweise zusätzliche Informationen für DAS, Advisor, Sentinel Control Center, Collector Builder und die Kommunikationsschicht erforderlich.

1. Ziehen Sie hinsichtlich Ihres speziellen Problems das Kundensupport-Portal zurate:
 - Handelt es sich um ein bekanntes Problem mit einer provisorischen Lösung?
 - Wird dieses Problem durch die aktuellste Patchversion bzw. das aktuelle HotFix beseitigt?
 - Ist die Beseitigung dieses Problems zurzeit mithilfe einer zukünftigen Version geplant?

2. Bestimmen Sie die Art des Problems.
 - Kann es reproduziert werden? Können die Schritte zur Reproduktion des Problems aufgezählt werden?
 - Durch welche Benutzeraktion (falls zutreffend) wird das Problem verursacht?
 - Tritt das Problem regelmäßig auf?
3. Bestimmen Sie den Schweregrad dieses Problems.
 - Kann das System weiterhin verwendet werden?
4. Halten Sie Informationen zur Umgebung und den Systemen bereit, die betroffen sind.
 - Welche Plattformen und Produktversionen sind betroffen?
 - Sind nichtstandardmäßige bzw. benutzerdefinierte Komponenten betroffen?
 - Handelt es sich um eine Umgebung mit hoher Ereignisrate?
 - Mit welcher Rate werden Ereignisse erfasst?
 - Wie lautet die Ereignisrate für das Einfügen in die Datenbank?
 - Wie viele gleichzeitige Benutzer gibt es?
 - Wird die Crystal-Berichtsfunktion verwendet? Wann werden Berichte ausgeführt?
 - Kommt die Korrelation zum Einsatz? Wie viele Regeln werden bereitgestellt?

Stellen Sie Konfigurationsdateien, Protokolldateien und Systeminformationen zusammen. Tragen Sie diese Informationen für den möglichen Wissensaustausch in der Zukunft zusammen. Informationen zum Speicherort von Protokolldateien finden Sie im Sentinel-Installationshandbuch, Kapitel 2 – Optimale Verfahren.
5. Überprüfen Sie den Zustand des Systems.
 - Können Sie sich bei Sentinel Console anmelden?
 - Werden Ereignisse erstellt und in die Datenbank eingefügt? (Wenn die Konfiguration noch gegeben ist, führen Sie SendOneEvent aus und suchen Sie nach den Ereignissen)
 - Werden Ereignisse in Sentinel Console angezeigt?
 - Können Ereignisse mithilfe der Schnellabfrage aus der Datenbank abgerufen werden?
 - Überprüfen Sie RAM-Auslastung, Speicherplatz, Vorgangsaktivität, CPU-Nutzung sowie Netzwerkkonnektivität der betroffenen Hosts.
 - Vergewissern Sie sich, dass alle erwarteten Sentinel-Vorgänge ausgeführt werden. In Skripts wie hp_checkprocess werden unter Solaris die relevanten Vorgänge und ihr Status aufgelistet. Der Microsoft Task-Manager kann in einer Windows-Umgebung verwendet werden.
 - Suchen Sie in den Unterverzeichnissen von ESEC_HOME nach Core-Dumps. Ermitteln Sie, für welche Vorgänge Core-Dumps erstellt werden. (cd \$ESEC_HOME, find . -name core -print)
 - Prüfen Sie auf sqlplus-Internetzugang. Prüfen Sie auf Tabellenbereiche.

- Stellen Sie sicher, dass Sonic Broker ausgeführt wird. Die Konnektivität kann über die Sonic-Verwaltungskonsole überprüft werden. Vergewissern Sie sich, dass die unterschiedlichen Verbindungen für Novell-Vorgänge aktiv sind. Stellen Sie sicher, dass der Start von Sonic nicht durch eine Sperrdatei verhindert wird. Optional können Sie über den Sonic-Port eine Telnet-Verbindung zu diesem Server herstellen (telnet sentinel.company.com 10012)
 - Überprüfen Sie, ob die Überwachung auf dem Server ausgeführt wird. (ps -ef | grep watchdog)
 - Vergewissern Sie sich, dass die Wizard-Vorgänge ordnungsgemäß funktionieren. Wird Collector Manager ausgeführt? Wird Collector Manager in Collector Builder bzw. in Sentinel Console als aktiv angegeben? Werden Collectors ausgeführt? Wie viele pro Computer? Welche Connectors werden verwendet (Datei, Vorgang, Syslog, Firewall, Ereignisprotokoll usw.)? Wie viele Betriebsressourcen beanspruchen sie?
6. Liegt ein Problem mit der Datenbank vor?
- Ist mit „sqlplus“ die Anmeldung bei der Datenbank möglich?
 - Ermöglicht die Datenbank die sqlplus-Anmeldung über das Novell-dba-Konto beim ESEC-Schema?
 - Kann eine Tabelle erfolgreich abgerufen werden?
 - Kann eine select-Anweisung für eine Datenbanktabelle erfolgreich ausgeführt werden?
 - Überprüfen Sie die JDBC-(Java Database Connectivity-)Treiber, ihren Speicherort sowie ihre Klassenpfadeinstellungen.
 - Bei Oracle: Ist die Partitionierung installiert (Eingabe „select * from v\$version;“) und wird sie verwendet?
 - Wird die Datenbank von einem Administrator verwaltet? Von einer anderen Person?
 - Wurde die Datenbank von diesem Administrator verändert?
 - Kommt SDM zum Verwalten der Partitionen sowie zum Archivieren/Löschen der Partitionen zum Einsatz, um in der Datenbank mehr Speicherplatz freizugeben?
 - Bei Verwendung von SDM: Welche Bezeichnung trägt die aktuelle Partition? PMAX?
7. Überprüfen Sie, ob die Produktumgebungseinstellungen richtig sind.
- Vergewissern Sie sich, dass Shell-Skripts für die Benutzeranmeldung, Umgebungsvariablen, Konfigurationen sowie Java Home-Einstellungen richtig sind.
 - Ist die Umgebungsvariable für die Ausführung der richtigen JVM (Java Virtual Machine) konfiguriert?
 - Vergewissern Sie sich, dass die entsprechenden Berechtigungen für die Ordner des installierten Produkts gewährt wurden.
 - Überprüfen Sie, ob Cron-Aufträge eingerichtet sind und zu Störungen der Funktionalität des Produkts führen.
 - Wenn das Produkt auf NFS-(Network File System-)Mounts installiert wurde, stellen Sie den ordnungsgemäßen Zustand von NFS-Mounts & NFS-/NIS-(Network Information System-)Services sicher.

8. Liegt möglicherweise ein Arbeitsspeicherleck vor?

- Rufen Sie die Statistiken dazu ab, wie schnell der Arbeitsspeicher belegt wird und von welchem Vorgang.
- Ermitteln Sie die Metrik des Ereignisdurchsatzes pro Collector.
- Führen Sie unter Solaris den prstat-Befehl aus. Auf diese Weise werden die Statistiken zur Vorgangslaufzeit abgerufen.
- Unter Windows gibt der Task-Manager Aufschluss über Vorgangsumfang und Handleanzahl.

Dieses Problem kann, falls nicht behoben, nun eskaliert werden. Mögliche Ergebnisse der Eskalation:

- Erweiterungen
- HotFixes
- Temporäre Lösungen

B

Einrichten des Sentinel-Service-Anmeldekontos als NT AUTHORITY\NetworkService

HINWEIS: Die Begriffe „Agent“ und „Collector“ sind austauschbar. Im Folgenden werden Agenten als „Collectors“ bezeichnet.

In diesem Dokument wird detailliert erläutert, wie das Sentinel-Service-Anmeldekonto anstatt als Domänenbenutzerkonto als NT AUTHORITY\NetworkService eingerichtet wird. Die Erfahrung hat gezeigt, dass dieser Vorgang nur auf der Windows 2003-Plattform möglich ist.

Ein Service muss sich bei einem Konto anmelden, um auf Ressourcen und Objekte des Betriebssystems zugreifen zu können. Wenn Sie ein Konto auswählen, das nicht zur Anmeldung als Service berechtigt ist, gewährt das Services-Snapin diesem Konto automatisch die Benutzerrechte, die erforderlich sind, um sich auf dem von Ihnen verwalteten Computer als Service anzumelden. Hierdurch wird jedoch nicht garantiert, dass der Service gestartet werden kann. Es empfiehlt sich, für die Benutzerkonten, die zur Anmeldung als Service verwendet werden, im entsprechenden Dialogfeld mit den Eigenschaften das Kontrollkästchen **Passwort läuft nie ab** zu aktivieren; zudem sollten diese Konten über ein komplexes Passwort verfügen. Wenn „Kontosperrungsrichtlinien“ aktiviert und das Konto gesperrt ist, kommt es zu einer Fehlfunktion des Service.

In der nachfolgenden Tabelle werden Serviceanmeldekonten und ihre Verwendung erläutert.

Anmeldekonto	Beschreibung
Lokales Systemkonto	Das lokale Systemkonto ist ein Konto mit umfassenden Berechtigungen, mit uneingeschränktem Zugriff auf das System, einschließlich des Verzeichnisservice für Domänencontroller. Wenn sich ein Service beim lokalen Systemkonto auf einem Domänencontroller anmeldet, kann dieser Service auf die gesamte Domäne zugreifen. Einige Dienste sind standardmäßig für die Anmeldung beim lokalen Systemkonto konfiguriert. Belassen Sie die standardmäßige Serviceeinstellung in jedem Fall unverändert.
	Das lokale Systemkonto ist ein vordefiniertes lokales Konto, mit dem ein Service gestartet und der Sicherheitskontext für diesen Service bereitgestellt wird. Der Name des Kontos lautet NT AUTHORITY\System. Für dieses Konto gibt es kein Passwort und sämtliche Passwortangaben, die Sie machen, werden ignoriert. Das lokale Systemkonto hat uneingeschränkten Zugriff auf das System, einschließlich des Verzeichnisservice für Domänencontroller. Da das lokale Systemkonto im Netzwerk als Computer agiert, kann es auch auf Netzwerkressourcen zugreifen.

Anmeldekonto	Beschreibung
Lokales Servicekonto	<p>Das lokale Servicekonto ist ein spezielles integriertes Konto, das mit einem authentifizierten Benutzerkonto vergleichbar ist. Das lokale Servicekonto weist dieselbe Zugriffsstufe auf Ressourcen und Objekte auf wie Mitglieder der Benutzer-Gruppe. Dieser eingeschränkte Zugriff trägt zum Schutz Ihres Systems für den Fall bei, dass einzelne Services oder Vorgänge gefährdet werden. Services, die unter dem lokalen Servicekonto ausgeführt werden, greifen als Null-Sitzung ohne Berechtigungsnachweise auf Netzwerkressourcen zu.</p> <p>Das lokale Servicekonto ist ein vordefiniertes lokales Konto, mit dem ein Service gestartet und der Sicherheitskontext für diesen Service bereitgestellt wird. Der Name des Kontos lautet NT AUTHORITY\LocalService. Das lokale Servicekonto weist eingeschränkten Zugriff auf den lokalen Computer und anonymen Zugriff auf Netzwerkressourcen auf.</p>
Netzwerkservicekonto	<p>Das Netzwerkservicekonto (NetworkService) ist ein spezielles integriertes Konto, das mit einem authentifizierten Benutzerkonto vergleichbar ist. Das Netzwerkservicekonto weist dieselbe Zugriffsstufe auf Ressourcen und Objekte auf wie Mitglieder der Benutzer-Gruppe. Dieser eingeschränkte Zugriff trägt zum Schutz Ihres Systems für den Fall bei, dass einzelne Services oder Vorgänge gefährdet werden. Services, die unter dem Netzwerkservicekonto ausgeführt werden, greifen unter Verwendung der Berechtigungsnachweise des Computerkontos auf Netzwerkressourcen zu.</p> <p>Das Netzwerkservicekonto (NetworkService) ist ein vordefiniertes lokales Konto, mit dem ein Service gestartet und der Sicherheitskontext für diesen Service bereitgestellt wird. Der Name des Kontos lautet NT AUTHORITY\NetworkService. Das Netzwerkservicekonto weist eingeschränkten Zugriff auf den lokalen Computer und authentifizierten Zugriff (als Computerkonto) auf Netzwerkressourcen auf.</p>

Die Ausführung eines Service im Kontext eines Benutzeranmeldekontos bringt folgende Nachteile mit sich:

1. Das Konto muss erstellt werden, bevor der Service ausgeführt werden kann. Wenn das Setup-Programm für den Service das Konto erstellt, muss Setup von einem Konto aus ausgeführt werden, das über ausreichende Administrationsberechtigungsnachweise zur Erstellung von Konten im Verzeichnisservice verfügt.
2. Servicekontonamen und -passwörter werden auf sämtlichen Computern gespeichert, auf denen der Service installiert wird. Wenn das Passwort für ein Servicekonto abläuft bzw. geändert wird, kann der Service auf diesem Computer erst gestartet werden, wenn das Passwort dem neuen Passwort für diesen Service entspricht. Es empfiehlt sich, das lokale Servicekonto bzw. das Netzwerkservicekonto anstelle eines Kontos zu verwenden, für das ein Passwort erforderlich ist: Auf diese Weise wird die Passwortverwaltung vereinfacht.

3. Wenn ein Servicekonto umbenannt, gesperrt, deaktiviert oder gelöscht wird, kann der Service auf diesem Computer erst gestartet werden, wenn das Konto zurückgesetzt wurde.

Aufgrund der oben genannten Nachteile hat Novell die Ausführung des Sentinel-Service über das NT AUTHORITY\NetworkService-Konto getestet. Das NT AUTHORITY\LocalService-Konto weist keine ausreichende Berechtigung für diesen Zweck auf, da DAS-Vorgänge mit dem Datenbankserver im Netzwerk kommunizieren müssen.

So richten Sie NT AUTHORITY\NetworkService als Anmeldekonto für den Sentinel-Service ein

Zur Einrichtung von NT AUTHORITY\NetworkService als Anmeldekonto für den Sentinel-Service müssen folgende Schritte durchgeführt werden:

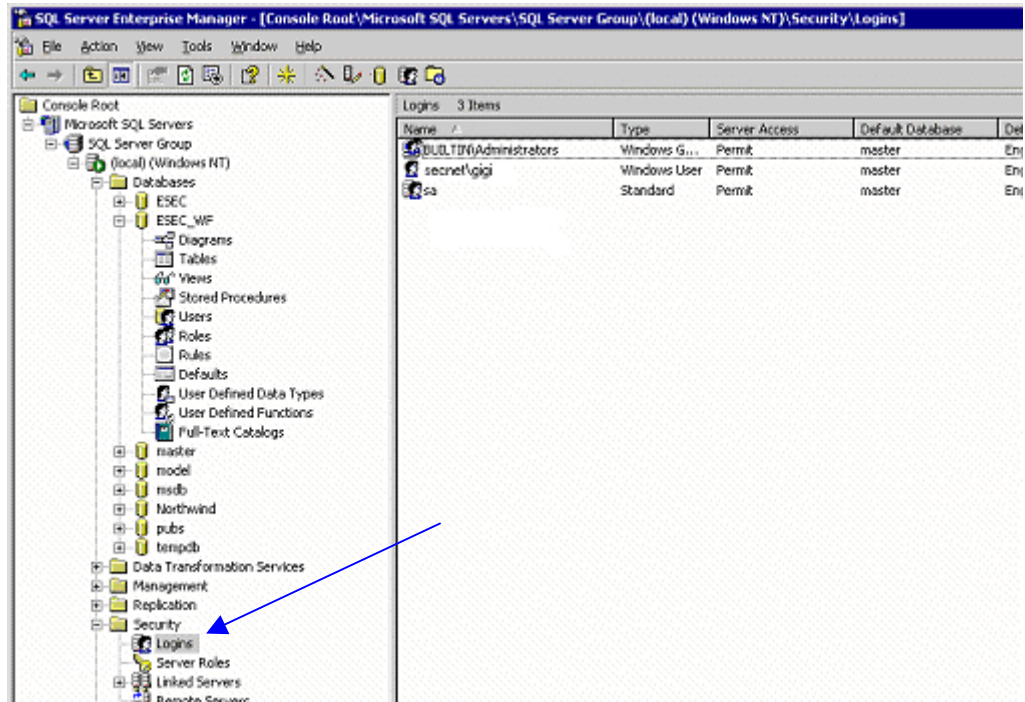
- Der Computer, auf dem der Sentinel-Service ausgeführt wird, muss als Anmeldekonto zu ESEC- und ESEC_WF-Datenbankinstanzen hinzugefügt werden (dieser Vorgang muss auf dem Computer mit der Datenbank erfolgen)
- Das Anmeldekonto für den Sentinel-Service muss in NT AUTHORITY\NetworkService geändert werden (dieser Vorgang muss auf dem Remote-Computer erfolgen)
- Der Sentinel-Start muss eingerichtet werden (dieser Vorgang muss auf dem Remote-Computer erfolgen)

Hinzufügen eines Sentinel-Service als Anmeldekonto zu ESEC- und ESEC_WF-DB-Instanzen

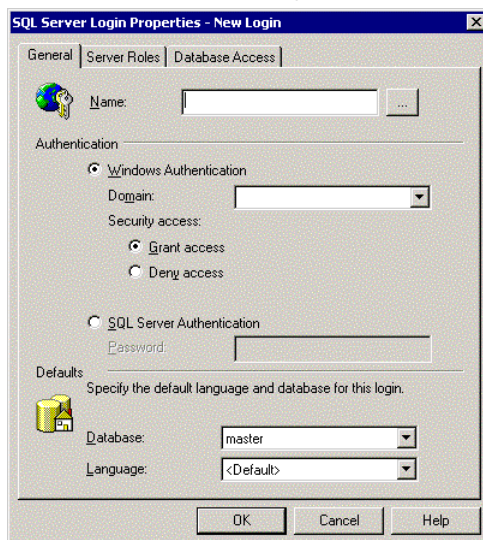
Hinzufügen einer Anmeldung eines Remote-Computers zum Datenbankserver

HINWEIS: Die nachfolgenden Schritte dienen zur Erläuterung der Schritte zum Hinzufügen von „secnet\case1“ als Anmeldung zum Datenbankserver.

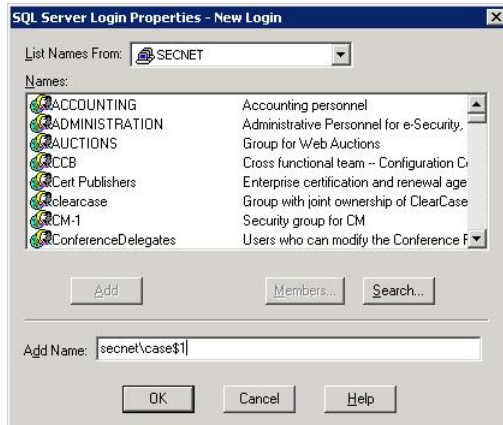
1. Rufen Sie SQL Server Enterprise Manager auf dem Datenbankcomputer auf. Erweitern Sie im Navigationsbereich unterhalb von „SQL Server Group“ (SQL Server-Gruppe) den Ordner „Security“ (Sicherheit) und markieren Sie den Eintrag „Logins“ (Anmeldungen).



2. Klicken Sie mit der rechten Maustaste auf *Logins* (Anmeldungen) und dann auf *New login...* (Neue Anmeldung...).

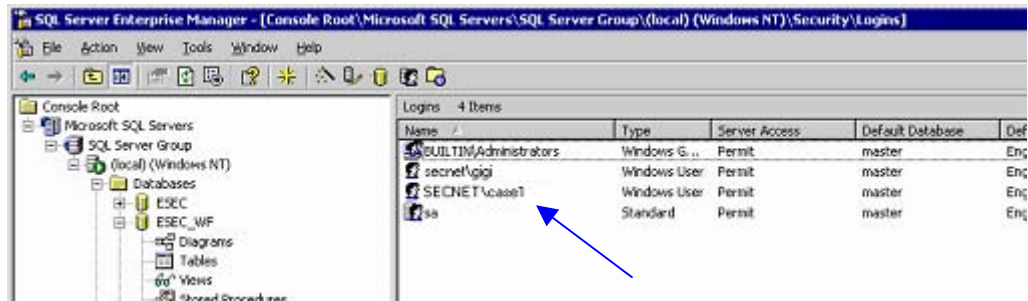


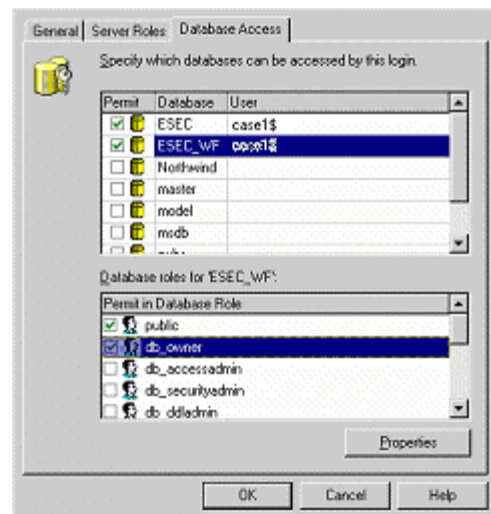
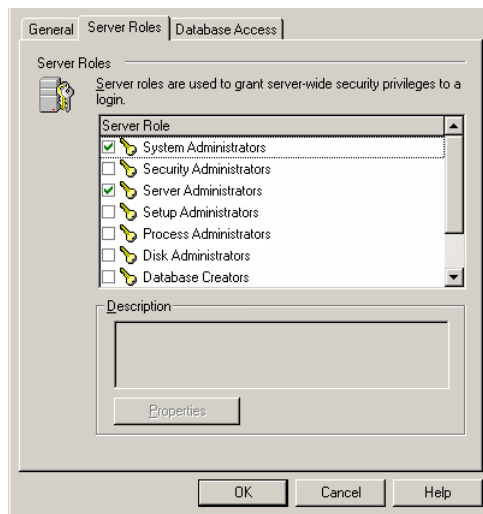
3. Wenn Sie beim Feld „Name“ (Name) auf die Schaltfläche zum Durchsuchen klicken, wird das nachfolgende Dialogfeld angezeigt.



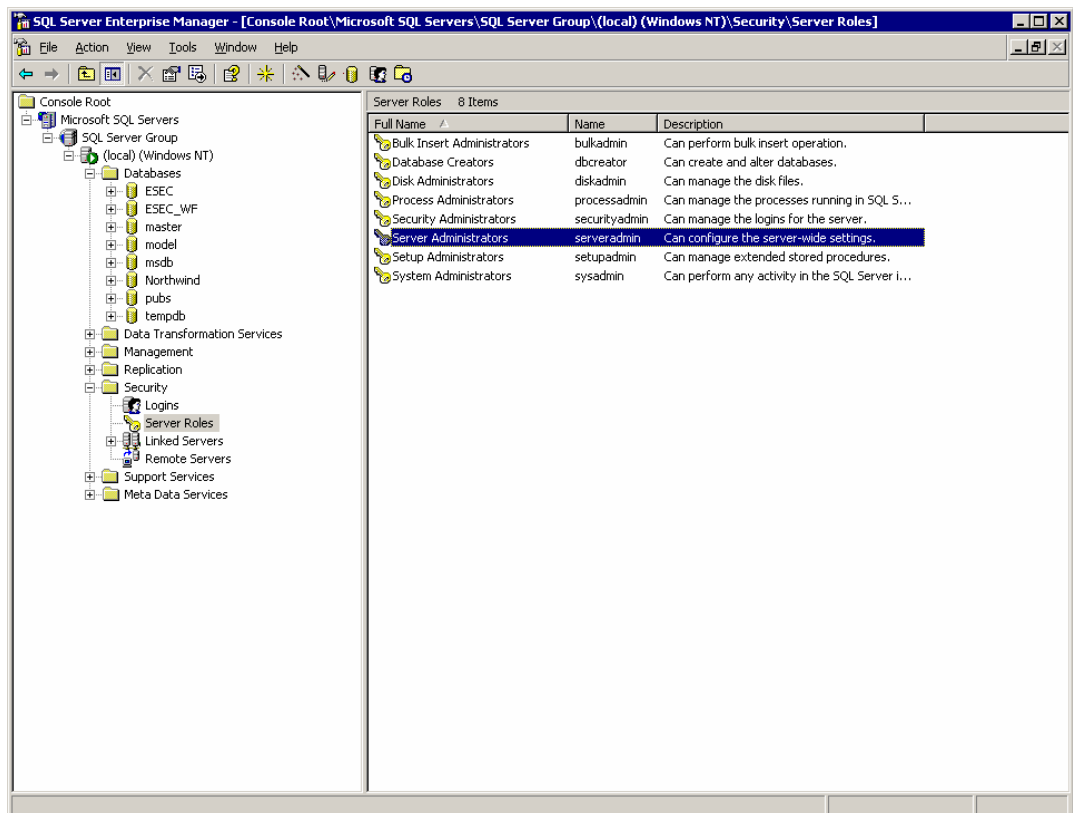
Geben Sie im Feld „Add Name“ (Namen hinzufügen) einen Domänennamen und einen Benutzernamen ein („secnet\case1\$“ dient als Beispiel). Hierbei handelt es sich um den Computer <domain name>\<name of machine>\$, den Sie als Anmeldung zum Datenbankserver hinzufügen. Klicken Sie auf **OK** (OK).

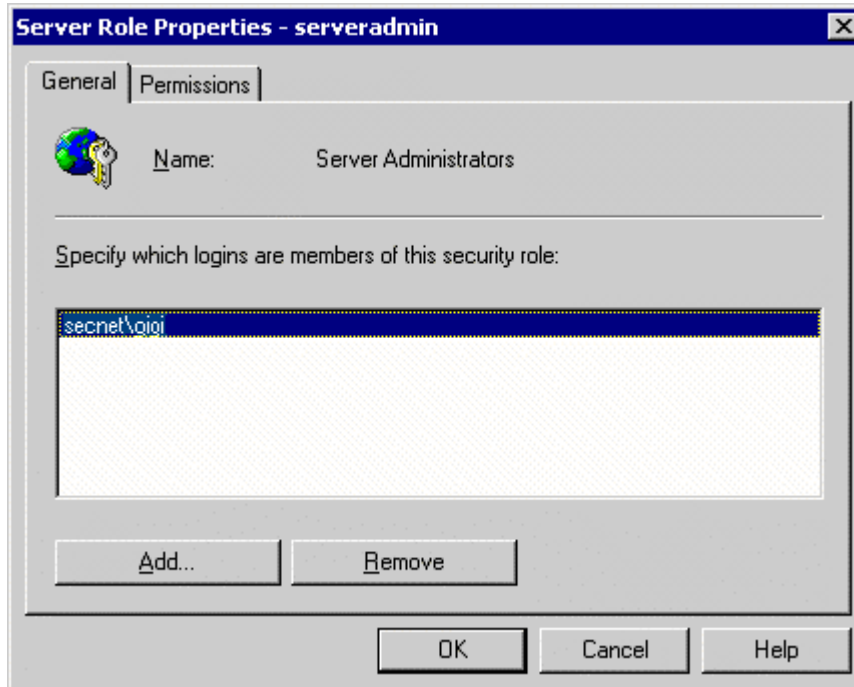
4. Klicken Sie mit der rechten Maustaste auf die Eigenschaften des Namens (des Computers [<domain name>\<name of machine>]), den Sie als Anmeldung zum Datenbankserver hinzufügen, um Serverfunktionen und Datenbankzugriff zu ändern. Wählen Sie „System Administrators“ (Systemadministratoren) und „Server Administrators“ (Serveradministratoren) als Serverfunktionen aus. Legen Sie für den ESEC-Zugriff „public“ (öffentlich) und „db_owner“ (db_owner) fest. Legen Sie für den ESEC_WF-Zugriff „public“ (öffentlich) und „db_owner“ (db_owner) fest.



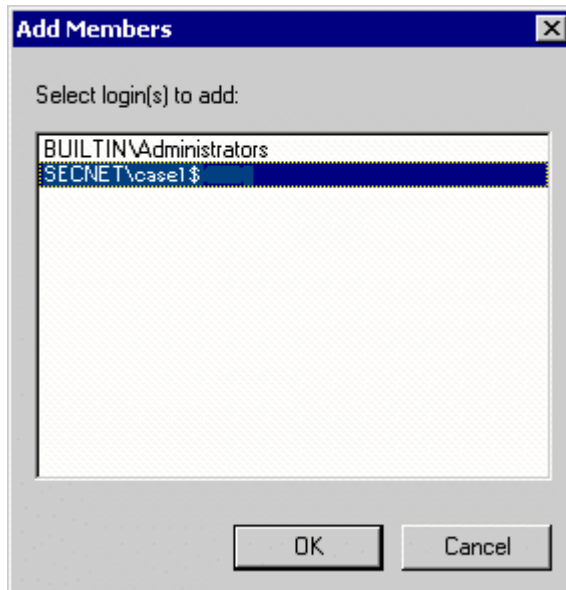


5. Markieren Sie unterhalb von „Server Roles“ (Serverfunktionen) den Eintrag „Server Administrators“ (Serveradministratoren), klicken Sie mit der rechten Maustaste und wählen Sie dann *Properties* (Eigenschaften).





6. Klicken Sie auf die Schaltfläche *Add* (Hinzufügen).

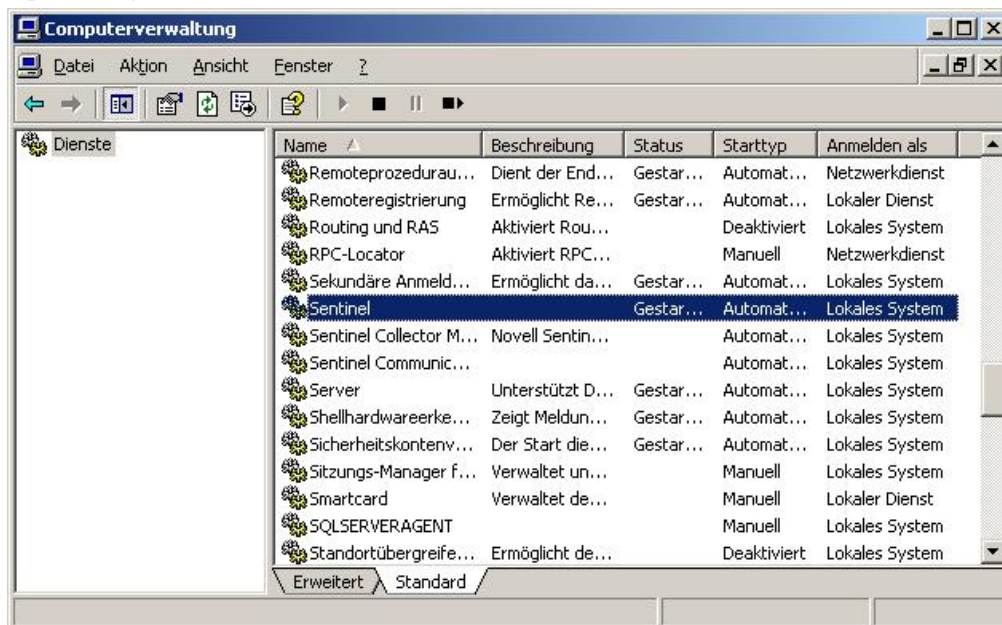


Mit „OK“ (OK) wird „Secnet\case1\$“ hinzugefügt.

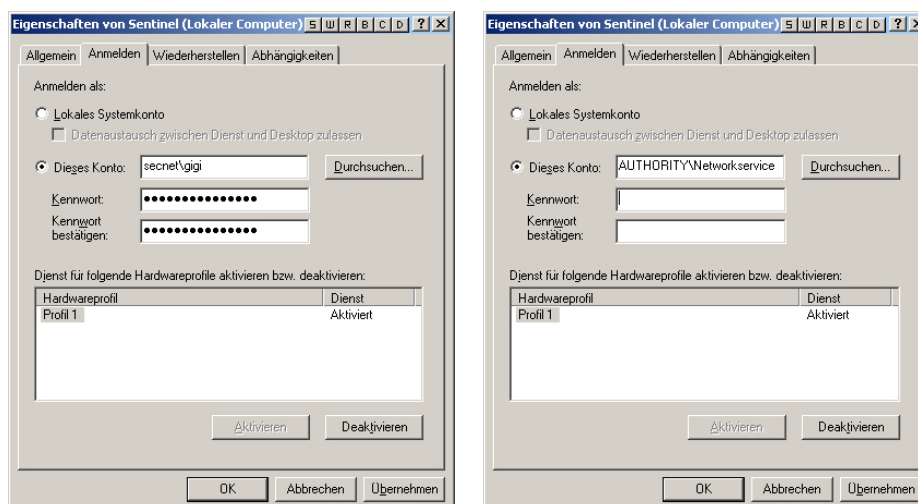
Ändern des Anmeldekontos des Sentinel-Service in NT AUTHORITY\NetworkService

Ändern der Anmeldung für Sentinel-Service in NT AUTHORITY\NetworkService

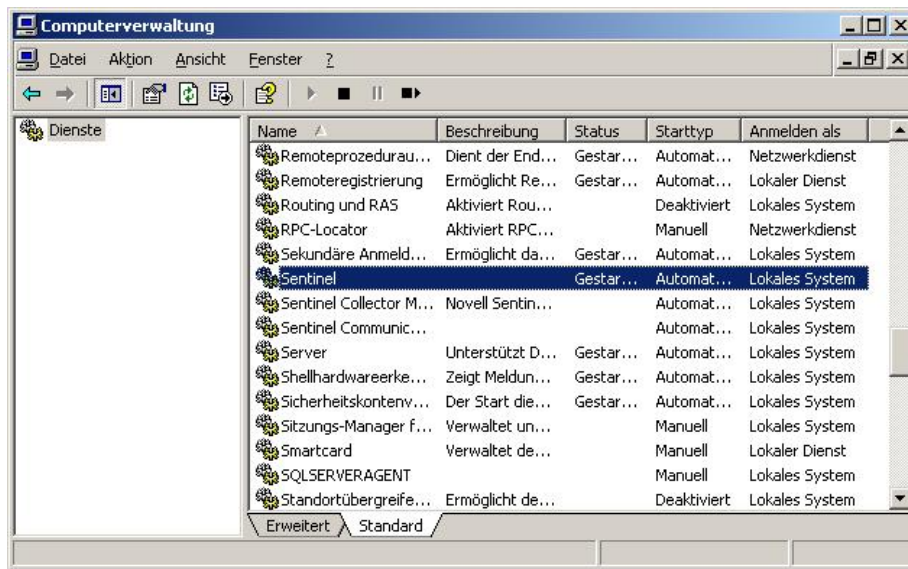
1. Wählen Sie auf dem Remote-Computer für die Verbindung mit der Datenbank die Optionsfolge *Start > Programme > Verwaltung > Dienste*.



2. Stoppen Sie den Sentinel-Service, klicken Sie mit der rechten Maustaste, wählen Sie *Eigenschaften* und aktivieren Sie dann die Registerkarte *Anmelden*.
3. Klicken Sie auf „Dieses Konto“ und geben Sie im zugehörigen Feld „NT AUTHORITY\NetworkService“ ein. Entfernen Sie eventuelle Einträge aus den Feldern „Passwort“ und „Passwortbestätigung“.



Klicken Sie auf **OK** (OK). Im Fenster für den Sentinel-Service sollte in der Spalte „Anmelden als“ „Netzwerkdienst“ angegeben sein.



Gewährleisten des erfolgreichen Starts des Sentinel-Services

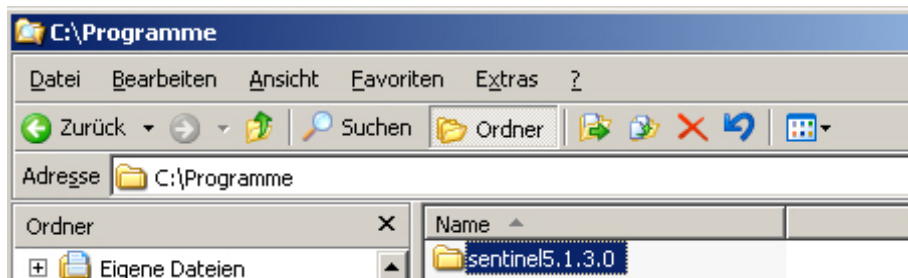
Damit der Sentinel-Service erfolgreich gestartet werden kann, sollte das NT AUTHORITY\NetworkService-Konto über Schreibberechtigung für %ESEC_HOME% verfügen. Laut Microsoft-Dokumentation verfügt das NetworkService-Konto über folgende Privilegien:

- SE_AUDIT_NAME
- SE_CHANGE_NOTIFY_NAME
- SE_UNDOCK_NAME
- Sämtliche Privilegien, die Benutzern und authentifizierten Benutzern zugewiesen wurden

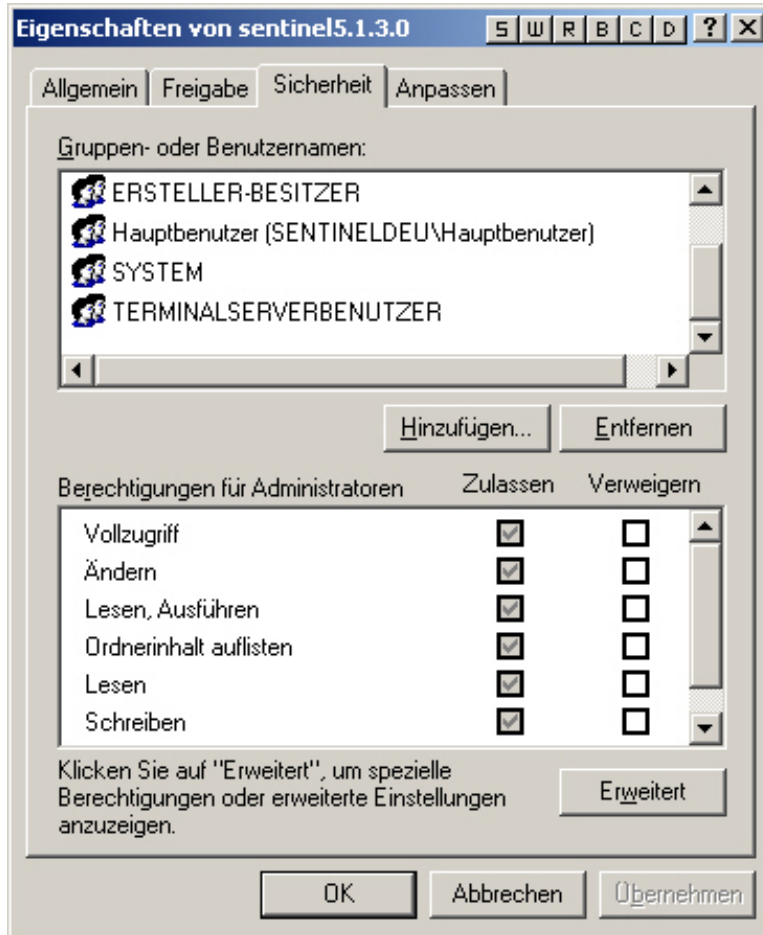
Sie müssen der Gruppe „Benutzer“ Schreibzugriff für %ESEC_HOME% gewähren.

Gewährleisten des erfolgreichen Starts des Sentinel-Services

1. Begeben Sie sich in Windows-Explorer zu %ESEC_HOME%.
2. Klicken Sie mit der rechten Maustaste auf den übergeordneten Sentinel-Ordner (er trägt normalerweise die Bezeichnung sentinel 5.1.3) > wählen Sie *Eigenschaften* > *Sicherheit*.



3. Markieren Sie die Gruppe „Benutzer“. Erteilen Sie die Berechtigung „Lesen, Ausführen“, „Ordnerinhalt auflisten“, „Lesen“ und „Schreiben“.



Klicken Sie auf *OK* (OK).

4. Starten Sie im Fenster mit den Diensten den Sentinel-Service neu.

C

Benutzer, Funktionen und Zugriffsberechtigungen der Sentinel-Datenbank

HINWEIS: Die Begriffe „Agent“ und „Collector“ sind austauschbar. Im Folgenden werden Agenten als „Collectors“ bezeichnet.

In diesem Dokument finden Sie eine detaillierte Aufschlüsselung der Sentinel-Datenbankbenutzer und -funktionen sowie deren jeweiligen Zugriffsberechtigungen.

Sentinel-Datenbankinstanz

ESEC

Benutzer:

- esecadm
- esecapp
- esecdba
- esecrpt
- Sonstige Benutzer

HINWEIS: Die oben aufgeführten Benutzer werden über „Benutzer-Manager“ erstellt. Detaillierte Zugriffsberechtigungen finden Sie im Abschnitt „Benutzer der Sentinel-Datenbank“.

Funktionen:

- ESEC_APP – Dieselbe Berechtigung wie db_owner
- ESEC_ETL – Diese Funktion wird zurzeit nicht verwendet; sie ist für eine künftige Aktualisierung bestimmt. Detaillierte Zugriffsberechtigungen finden Sie im Abschnitt [Funktionen in der Sentinel-Datenbank](#).
- ESEC_USER – Detaillierte Zugriffsberechtigungen finden Sie im Abschnitt [Funktionen in der Sentinel-Datenbank](#).

ESEC_WF

- Benutzer: esecapp – Detaillierte Zugriffsberechtigungen finden Sie im Abschnitt [Benutzer der Sentinel-Datenbank](#).
- Funktionen: ESEC_APP – Detaillierte Zugriffsberechtigungen finden Sie im Abschnitt [Funktionen in der Sentinel-Datenbank](#).

Benutzer der Sentinel-Datenbank

Zusammenfassung

Benutzername	Gruppenname	Anmeldename	DB-Standardname
esecadm	ESEC_USER	esecadm	ESEC
esecapp	ESEC_APP	esecapp	ESEC
esecapp	ESEC_ETL	esecapp	ESEC
esecdba	db_owner	esecdba	ESEC
esecrpt	ESEC_USER	esecrpt	ESEC

esecadm

Anmeldename	DB-Name	Benutzername	Benutzer von Alias
esecadm	ESEC	ESEC_USER	MemberOf
esecadm	ESEC	esecadm	User

esecapp

Anmeldename	DB-Name	Benutzername	Benutzer von Alias
esecapp	ESEC	ESEC_APP	MemberOf
esecapp	ESEC	ESEC_ETL	MemberOf
esecapp	ESEC	esecapp	User
esecapp	ESEC_WF	ESEC_APP	MemberOf
esecapp	ESEC_WF	esecapp	User

esecdba

Anmeldename	DB-Name	Benutzername	Benutzer von Alias
esecdba	ESEC	db_owner	MemberOf
esecdba	ESEC	esecdba	User

esecrpt

Anmeldename	DB-Name	Benutzername	Benutzer von Alias
esecrpt	ESEC	ESEC_USER	MemberOf
esecrpt	ESEC	esecrpt	User

Funktionen in der Sentinel-Datenbank

Zusammenfassung

- ESEC_APP – Es handelt sich um eine Datenbankfunktion für ESEC und ESEC_WF. Sie verfügt für die ESEC-Instanz über dieselbe Berechtigung wie db_owner. Details zu Berechtigungen finden Sie im Abschnitt [ESEC_APP](#).
- ESEC_ETL – Es handelt sich um eine Datenbankfunktion für die ESEC-Instanz. Sie wird zurzeit nicht verwendet und ist für künftige Entwicklungen vorgesehen. Detaillierte Zugriffsberechtigungen finden Sie im Abschnitt [Sentinel Database Roles](#).
- ESEC_USER – Eine Funktion für die ESEC-Instanz. Detaillierte Zugriffsberechtigungen finden Sie im Abschnitt [Sentinel Database Roles](#).

ESEC_APP

Für die ESEC-Instanz verfügt ESEC_APP über dieselbe Berechtigung wie db_owner. ESEC_APP führt die Aktivitäten aller Datenbankfunktionen sowie weitere Wartungs- und Konfigurationsaktivitäten in der Datenbank durch. Die Berechtigungen dieser Funktion umfassen alle der weiteren festen Datenbankfunktionen.

Für die ESEC_WF-Instanz ist dies die Berechtigung für die ESEC_APP-Funktion.

Funktionsname	Objektname	Aktion	Typ
ESEC_APP	Activities	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	Activities	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	Activities	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	Activities	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	ActivityData	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ActivityData	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ActivityData	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ActivityData	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	ActivityStateEventAudits	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ActivityStateEventAudits	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ActivityStateEventAudits	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ActivityStateEventAudits	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	ActivityStates	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ActivityStates	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ActivityStates	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ActivityStates	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	AndJoinTable	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	AndJoinTable	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	AndJoinTable	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	AndJoinTable	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	AssignmentEventAudits	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	AssignmentEventAudits	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	AssignmentEventAudits	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	AssignmentEventAudits	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	AssignmentsTable	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	AssignmentsTable	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	AssignmentsTable	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	AssignmentsTable	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	Counters	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	Counters	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	Counters	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	Counters	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	CreateProcessEventAudits	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	CreateProcessEventAudits	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	CreateProcessEventAudits	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	CreateProcessEventAudits	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	DataEventAudits	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	DataEventAudits	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	DataEventAudits	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	DataEventAudits	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	Deadlines	193 AUSWÄHLEN	U Benutzertabelle

Funktionsname	Objektname	Aktion	Typ
ESEC_APP	Deadlines	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	Deadlines	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	Deadlines	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	EventTypes	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	EventTypes	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	EventTypes	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	EventTypes	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	GroupGroupTable	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	GroupGroupTable	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	GroupGroupTable	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	GroupGroupTable	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	GroupTable	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	GroupTable	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	GroupTable	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	GroupTable	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	GroupUser	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	GroupUser	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	GroupUser	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	GroupUser	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	GroupUserPackLevelParticipant	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	GroupUserPackLevelParticipant	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	GroupUserPackLevelParticipant	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	GroupUserPackLevelParticipant	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	GroupUserPackLevelParticipant	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	GroupUserPackLevelParticipant	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	GroupUserPackLevelParticipant	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	GroupUserPackLevelParticipant	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	LockTable	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	LockTable	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	LockTable	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	LockTable	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	NewEventAuditData	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	NewEventAuditData	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	NewEventAuditData	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	NewEventAuditData	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	NextXPDLVersions	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	NextXPDLVersions	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	NextXPDLVersions	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	NextXPDLVersions	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	NormalUser	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	NormalUser	195 EINFÜGEN	U Benutzertabelle

Funktionsname	Objektname	Aktion	Typ
ESEC_APP	NormalUser	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	NormalUser	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	ObjectId	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ObjectId	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ObjectId	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ObjectId	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	OldEventAuditData	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	OldEventAuditData	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	OldEventAuditData	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	OldEventAuditData	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	PackLevelParticipant	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	PackLevelParticipant	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	PackLevelParticipant	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	PackLevelParticipant	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	PackLevelXPDLApp	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	PackLevelXPDLApp	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	PackLevelXPDLApp	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	PackLevelXPDLApp	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	PackLevelXPDLAppTAA		
ESEC_APP	ppDetail	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	PackLevelXPDLAppTAA		
ESEC_APP	ppDetail	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	PackLevelXPDLAppTAA		
ESEC_APP	ppDetail	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	PackLevelXPDLAppTAA		
ESEC_APP	ppDetail	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	PackLevelXPDLAppTAA		
ESEC_APP	ppDetailUsr	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	PackLevelXPDLAppTAA		
ESEC_APP	ppDetailUsr	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	PackLevelXPDLAppTAA		
ESEC_APP	ppDetailUsr	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	PackLevelXPDLAppTAA		
ESEC_APP	ppDetailUsr	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	PackLevelXPDLAppTAA		
ESEC_APP	ppUser	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	PackLevelXPDLAppTAA		
ESEC_APP	ppUser	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	PackLevelXPDLAppTAA		
ESEC_APP	ppUser	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	PackLevelXPDLAppTAA		
ESEC_APP	ppUser	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	PackLevelXPDLAppTool		
ESEC_APP	AgentApp	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	PackLevelXPDLAppTool		
ESEC_APP	AgentApp	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	PackLevelXPDLAppTool		
ESEC_APP	AgentApp	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	PackLevelXPDLAppTool	197 AKTUALISIEREN	U Benutzertabelle

Funktionsname	Objektname	Aktion	Typ
	AgentApp		
ESEC_APP	ProcessData	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ProcessData	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ProcessData	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ProcessData	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	ProcessDefinitions	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ProcessDefinitions	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ProcessDefinitions	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ProcessDefinitions	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	Processes	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	Processes	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	Processes	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	Processes	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	ProcessRequesters	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ProcessRequesters	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ProcessRequesters	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ProcessRequesters	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	ProcessStateEventAudits	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ProcessStateEventAudits	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ProcessStateEventAudits	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ProcessStateEventAudits	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	ProcessStates	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ProcessStates	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ProcessStates	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ProcessStates	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	ProcLevelParticipant	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ProcLevelParticipant	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ProcLevelParticipant	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ProcLevelParticipant	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	ProcLevelXPDLApp	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ProcLevelXPDLApp	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ProcLevelXPDLApp	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ProcLevelXPDLApp	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	ProcLevelXPDLAppTAA		
ESEC_APP	ppDetail	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ProcLevelXPDLAppTAA		
ESEC_APP	ppDetail	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ProcLevelXPDLAppTAA		
ESEC_APP	ppDetail	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ProcLevelXPDLAppTAA		
ESEC_APP	ppDetail	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	ProcLevelXPDLAppTAA		
ESEC_APP	ppDetailUsr	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ProcLevelXPDLAppTAA		
ESEC_APP	ppDetailUsr	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ProcLevelXPDLAppTAA		
ESEC_APP	ppDetailUsr	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ProcLevelXPDLAppTAA		
ESEC_APP	ppDetailUsr	197 AKTUALISIEREN	U Benutzertabelle

Funktionsname	Objektname	Aktion	Typ
ESEC_APP	ProcLevelXPDLAppTAA ppUser	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ProcLevelXPDLAppTAA ppUser	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ProcLevelXPDLAppTAA ppUser	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ProcLevelXPDLAppTAA ppUser	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	ProcLevelXPDLAppTool AgentApp	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ProcLevelXPDLAppTool AgentApp	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ProcLevelXPDLAppTool AgentApp	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ProcLevelXPDLAppTool AgentApp	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	ResourcesTable	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ResourcesTable	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ResourcesTable	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ResourcesTable	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	StateEventAudits	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	StateEventAudits	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	StateEventAudits	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	StateEventAudits	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	ToolAgentApp	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ToolAgentApp	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ToolAgentApp	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ToolAgentApp	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	ToolAgentAppDetail	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ToolAgentAppDetail	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ToolAgentAppDetail	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ToolAgentAppDetail	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	ToolAgentAppDetailUser	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ToolAgentAppDetailUser	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ToolAgentAppDetailUser	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ToolAgentAppDetailUser	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	ToolAgentAppUser	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ToolAgentAppUser	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ToolAgentAppUser	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ToolAgentAppUser	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	ToolAgentUser	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	ToolAgentUser	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	ToolAgentUser	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	ToolAgentUser	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	UserGroupTable	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	UserGroupTable	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	UserGroupTable	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	UserGroupTable	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	UserPackLevelParticipant	193 AUSWÄHLEN	U Benutzertabelle

Funktionsname	Objektname	Aktion	Typ
ESEC_APP	UserPackLevelParticipant	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	UserPackLevelParticipant	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	UserPackLevelParticipant	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	UserProcLevelParticipant	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	UserProcLevelParticipant	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	UserProcLevelParticipant	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	UserProcLevelParticipant	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	UserTable	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	UserTable	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	UserTable	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	UserTable	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	XPDLApplicationPackage	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	XPDLApplicationPackage	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	XPDLApplicationPackage	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	XPDLApplicationPackage	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	XPDLApplicationProcess	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	XPDLApplicationProcess	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	XPDLApplicationProcess	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	XPDLApplicationProcess	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	XPDLData	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	XPDLData	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	XPDLData	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	XPDLData	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	XPDLHistory	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	XPDLHistory	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	XPDLHistory	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	XPDLHistory	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	XPDLHistoryData	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	XPDLHistoryData	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	XPDLHistoryData	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	XPDLHistoryData	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	XPDLParticipantPackage	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	XPDLParticipantPackage	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	XPDLParticipantPackage	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	XPDLParticipantPackage	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	XPDLParticipantProcess	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	XPDLParticipantProcess	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	XPDLParticipantProcess	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	XPDLParticipantProcess	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	XPDLReferences	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	XPDLReferences	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	XPDLReferences	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	XPDLReferences	197 AKTUALISIEREN	U Benutzertabelle
ESEC_APP	XPDLs	193 AUSWÄHLEN	U Benutzertabelle
ESEC_APP	XPDLs	195 EINFÜGEN	U Benutzertabelle
ESEC_APP	XPDLs	196 LÖSCHEN	U Benutzertabelle
ESEC_APP	XPDLs	197 AKTUALISIEREN	U Benutzertabelle

ESEC_ETL

Funktionsname	Objektname	Aktion	Typ
ESEC_ETL	ACTVY	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ACTVY_NAMESPACE	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ACTVY_PARM	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ACTVY_REF	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ACTVY_REF_PARM_VAL	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ADV_ALERT	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ADV_ALERT_CVE	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ADV_ALERT_PRODUCT	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ADV_ATTACK	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ADV_ATTACK_ALERT	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ADV_ATTACK_CVE	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ADV_ATTACK_MAP	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ADV_ATTACK_PLUGIN	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ADV_CREDIBILITY	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ADV_FEED	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ADV_PRODUCT	193 AUSWÄHLEN	U Benutzertabelle
	ADV_PRODUCT_SERVICE_P		
ESEC_ETL	ACK	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ADV_PRODUCT_VERSION	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ADV_SEVERITY	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ADV_SUBALERT	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ADV_URGENCY	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ADV_VENDOR	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ADV_VULN_PRODUCT	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ANNOTATIONS	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ASSET	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ASSET_CTGRY	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ASSET_HOSTNAME	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ASSET_IP	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ASSET_LOC	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ASSET_VAL_LKUP	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ASSET_X_ENTITY_X_ROLE	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ASSOCIATIONS	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ATTACHMENTS	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	CONFIGS	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	CONTACTS	193 AUSWÄHLEN	U Benutzertabelle
	CORRELATED_EVENTS_P_M		
ESEC_ETL	AX	193 AUSWÄHLEN	U Benutzertabelle
	CORRELATED_EVENTS_P_M		
ESEC_ETL	IN	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	CRIT_LKUP	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	CUST	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ENTITY_TYP_LKUP	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ENV_IDENTITY_LKUP	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ESEC_ARCHIVE_CONFIG	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ESEC_ARCHIVE_LOG_FILES	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ESEC_ARCHIVE_LOGS	193 AUSWÄHLEN	U Benutzertabelle

Funktionsname	Objektname	Aktion	Typ
ESEC_ETL	ESEC_DB_PATCHES	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ESEC_DB_VERSION	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ESEC_DISPLAY	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ESEC_PARTITION_CONFIG	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ESEC_PARTITIONS_TEMP	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ESEC_PORT_REFERENCE	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ESEC_PROTOCOL_REFEREN CE	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ESEC_SDM_LOCK	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ESEC_SEQUENCE	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVENTS_P_MAX	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVENTS_P_MIN	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVT_AGENT	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVT_ASSET	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVT_DEST_EVT_NAME_SMR Y_1_P_MAX	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVT_DEST_EVT_NAME_SMR Y_1_P_MAX	195 EINFÜGEN	U Benutzertabelle
ESEC_ETL	EVT_DEST_EVT_NAME_SMR Y_1_P_MAX	196 LÖSCHEN	U Benutzertabelle
ESEC_ETL	EVT_DEST_EVT_NAME_SMR Y_1_P_MAX	197 AKTUALISIEREN	U Benutzertabelle
ESEC_ETL	EVT_DEST_EVT_NAME_SMR Y_1_P_MIN	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVT_DEST_SMRY_1_P_MAX	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVT_DEST_SMRY_1_P_MAX	195 EINFÜGEN	U Benutzertabelle
ESEC_ETL	EVT_DEST_SMRY_1_P_MAX	196 LÖSCHEN	U Benutzertabelle
ESEC_ETL	EVT_DEST_SMRY_1_P_MAX	197 AKTUALISIEREN	U Benutzertabelle
ESEC_ETL	EVT_DEST_SMRY_1_P_MIN	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVT_DEST_TXNMY_SMRY_1 _P_MAX	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVT_DEST_TXNMY_SMRY_1 _P_MAX	195 EINFÜGEN	U Benutzertabelle
ESEC_ETL	EVT_DEST_TXNMY_SMRY_1 _P_MAX	196 LÖSCHEN	U Benutzertabelle
ESEC_ETL	EVT_DEST_TXNMY_SMRY_1 _P_MAX	197 AKTUALISIEREN	U Benutzertabelle
ESEC_ETL	EVT_DEST_TXNMY_SMRY_1 _P_MIN	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVT_NAME	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVT_NAME	195 EINFÜGEN	U Benutzertabelle
ESEC_ETL	EVT_NAME	196 LÖSCHEN	U Benutzertabelle
ESEC_ETL	EVT_NAME	197 AKTUALISIEREN	U Benutzertabelle
ESEC_ETL	EVT_PORT_SMRY_1_P_MAX	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVT_PORT_SMRY_1_P_MAX	195 EINFÜGEN	U Benutzertabelle
ESEC_ETL	EVT_PORT_SMRY_1_P_MAX	196 LÖSCHEN	U Benutzertabelle
ESEC_ETL	EVT_PORT_SMRY_1_P_MAX	197 AKTUALISIEREN	U Benutzertabelle
ESEC_ETL	EVT_PORT_SMRY_1_P_MIN	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVT_PRTCL	193 AUSWÄHLEN	U Benutzertabelle

Funktionsname	Objektname	Aktion	Typ
ESEC_ETL	EVT_RSRC	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVT_SEV_SMRY_1_P_MAX	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVT_SEV_SMRY_1_P_MAX	195 EINFÜGEN	U Benutzertabelle
ESEC_ETL	EVT_SEV_SMRY_1_P_MAX	196 LÖSCHEN	U Benutzertabelle
ESEC_ETL	EVT_SEV_SMRY_1_P_MAX	197 AKTUALISIEREN	U Benutzertabelle
ESEC_ETL	EVT_SEV_SMRY_1_P_MIN	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVT_SRC_SMRY_1_P_MAX	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVT_SRC_SMRY_1_P_MAX	195 EINFÜGEN	U Benutzertabelle
ESEC_ETL	EVT_SRC_SMRY_1_P_MAX	196 LÖSCHEN	U Benutzertabelle
ESEC_ETL	EVT_SRC_SMRY_1_P_MAX	197 AKTUALISIEREN	U Benutzertabelle
ESEC_ETL	EVT_SRC_SMRY_1_P_MIN	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVT_TXNMY	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVT_USR	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	EVT_USR	195 EINFÜGEN	U Benutzertabelle
ESEC_ETL	EVT_USR	196 LÖSCHEN	U Benutzertabelle
ESEC_ETL	EVT_USR	197 AKTUALISIEREN	U Benutzertabelle
ESEC_ETL	EXT_DATA	193 AUSWÄHLEN	U Benutzertabelle
	HIST_CORRELATED_EVENT		
ESEC_ETL	S_P_MAX	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	HIST_EVENTS_P_MAX	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	IMAGES	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	INCIDENTS	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	INCIDENTS_ASSETS	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	INCIDENTS_EVENTS	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	INCIDENTS_VULN	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	L_STAT	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	LOGS	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	MD_CONFIG	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	MD_EVT_FILE_STS	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	MD_EVT_FILE_STS	195 EINFÜGEN	U Benutzertabelle
ESEC_ETL	MD_EVT_FILE_STS	196 LÖSCHEN	U Benutzertabelle
ESEC_ETL	MD_EVT_FILE_STS	197 AKTUALISIEREN	U Benutzertabelle
ESEC_ETL	MD_SMRY_STS	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	MD_SMRY_STS	195 EINFÜGEN	U Benutzertabelle
ESEC_ETL	MD_SMRY_STS	196 LÖSCHEN	U Benutzertabelle
ESEC_ETL	MD_SMRY_STS	197 AKTUALISIEREN	U Benutzertabelle
ESEC_ETL	MD_VIEW_CONFIG	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	NETWORK_IDENTITY_LKUP	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	OBJ_STORE	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ORGANIZATION	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	PERSON	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	PHYSICAL_ASSET	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	PRDT	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	ROLE_LKUP	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	SENSITIVITY_LKUP	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	STATES	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	USERS	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	VNDR	193 AUSWÄHLEN	U Benutzertabelle

Funktionsname	Objektname	Aktion	Typ
ESEC_ETL	VULN	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	VULN_CODE	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	VULN_INFO	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	VULN_RSRC	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	VULN_RSRC_SCAN	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	VULN_SCAN	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	VULN_SCAN_VULN	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	VULN_SCANNER	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	WORKFLOW_DEF	193 AUSWÄHLEN	U Benutzertabelle
ESEC_ETL	WORKFLOW_INFO	193 AUSWÄHLEN	U Benutzertabelle

ESEC_USER

Funktionsname	Objektname	Aktion	Typ
ESEC_USER	ADV_ALERT_CVE_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ADV_ALERT_PRODUCT_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ADV_ALERT_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ADV_ATTACK_ALERT_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ADV_ATTACK_CVE_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ADV_ATTACK_MAP_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ADV_ATTACK_PLUGIN_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ADV_ATTACK_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ADV_CREDIBILITY_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ADV_FEED_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ADV_PRODUCT_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ADV_PRODUCT_SERVICE_PACK_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ADV_PRODUCT_VERSION_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ADV_SEVERITY_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ADV_SUBALERT_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ADV_URGENCY_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ADV_VENDOR_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ADV_VULN_PRODUCT_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ANNOTATIONS_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ASSET_CATEGORY_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ASSET_HOSTNAME_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ASSET_IP_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ASSET_LOCATION_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ASSET_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ASSET_VALUE_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ASSET_X_ENTITY_X_ROLE_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ASSOCIATIONS_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ATTACHMENTS_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	CONFIGS_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	CONTACTS_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	CORRELATED_EVENTS	193 AUSWÄHLEN	V Ansicht
ESEC_USER	CORRELATED_EVENTS_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	CORRELATED_EVENTS_RPT_V1	193 AUSWÄHLEN	V Ansicht
ESEC_USER	CRITICALITY_RPT_V	193 AUSWÄHLEN	V Ansicht

Funktionsname	Objektname	Aktion	Typ
ESEC_USER	CUST_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ENTITY_TYPE_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ENV_IDENTITY_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ESEC_DISPLAY_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ESEC_PORT_REFERENCE_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ESEC_PROTOCOL_REFERENCE_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ESEC_SEQUENCE_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	esec_toBase	224 AUSFÜHREN	NULL
ESEC_USER	esec_toDecimal	224 AUSFÜHREN	NULL
ESEC_USER	esec_toIpChar	224 AUSFÜHREN	NULL
ESEC_USER	EVENTS	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVENTS_ALL_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVENTS_ALL_RPT_V1	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVENTS_ALL_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVENTS_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVENTS_RPT_V1	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVENTS_RPT_V2	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVT_AGENT_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVT_ASSET_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVT_DEST_EVT_NAME_SMRY_1	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVT_DEST_EVT_NAME_SMRY_1_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVT_DEST_SMRY_1	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVT_DEST_SMRY_1_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVT_DEST_TXNMY_SMRY_1	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVT_DEST_TXNMY_SMRY_1_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVT_NAME_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVT_PORT_SMRY_1	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVT_PORT_SMRY_1_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVT_PRTCL_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVT_RSRC_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVT_SEV_SMRY_1	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVT_SEV_SMRY_1_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVT_SRC_SMRY_1	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVT_SRC_SMRY_1_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVT_TXNMY_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EVT_USR_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	EXTERNAL_DATA_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	HIST_CORRELATED_EVENTS	193 AUSWÄHLEN	V Ansicht
ESEC_USER	HIST_CORRELATED_EVENTS_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	HIST_EVENTS	193 AUSWÄHLEN	V Ansicht
ESEC_USER	HIST_EVENTS_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	IMAGES_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	INCIDENTS_ASSETS_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	INCIDENTS_EVENTS_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	INCIDENTS_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	INCIDENTS_VULN_RPT_V	193 AUSWÄHLEN	V Ansicht

Funktionsname	Objektname	Aktion	Typ
ESEC_USER	L_STAT_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	LOGS_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	NETWORK_IDENTITY_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ORGANIZATION_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	PERSON_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	PHYSICAL_ASSET_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	PRODUCT_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	ROLE_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	SENSITIVITY_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	STATES_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	UNASSIGNED_INCIDENTS_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	USERS_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	VENDOR_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	VULN_CALC_SEVERITY_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	VULN_CODE_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	VULN_INFO_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	VULN_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	VULN_RSRC_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	VULN_RSRC_SCAN_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	VULN_SCAN_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	VULN_SCAN_VULN_RPT_V	193 AUSWÄHLEN	V Ansicht
ESEC_USER	VULN_SCANNER_RPT_V	193 AUSWÄHLEN	V Ansicht

Sentinel Server-Funktionen

Serverfunktion	Beschreibung	Sentinel-Benutzer
sysadmin	Systemadministratoren	esecdba
securityadmin	Sicherheitsadministratoren	esecapp
serveradmin	Serveradministratoren	esecdba
setupadmin	Einrichtungsadministratoren	
processadmin	Vorgangsadministratoren	
diskadmin	Datenträgeradministratoren	
dbcreator	Datenbankersteller	
bulkadmin	Masseneinfügingsadministratoren	

Windows-Domänenauthentifizierung: DB-Benutzer und -Berechtigungen

Ein Domänenbenutzer wird gemäß der Konfiguration zum Installationszeitpunkt mit dem esecadm-, esecapp-, esecdba- und esecrpt-Benutzer verknüpft. Diese Domänenbenutzer verfügen über dieselben Berechtigungen wie in den vorangegangenen Abschnitten angegeben.

D

Tabellen mit Sentinel-Serviceberechtigungen

HINWEIS: Die Begriffe „Agent“ und „Collector“ sind austauschbar. Im Folgenden werden Agenten als „Collectors“ bezeichnet.

Sentinel Server (Correlation Engine)

Sentinel-Komponente	Sentinel-Anwendung	Sentinel-Service	Sentinel-Vorgang	Funktionszusammenfassung	Erforderliche Berechtigungen	Erläuterung der Berechtigung
Sentinel Server	-	Sentinel/WatchDog.exe	correlation_engine.exe	Beim Correlation Engine-Prozess (correlation_engine) werden Ereignisse vom Wizard Collector Manager empfangen und korrelierte Ereignisse auf der Grundlage benutzerdefinierter Korrelationsregeln veröffentlicht.	Netzwerkzugriff; Lesezugriff auf geänderte Konfigurationsdateien.	Die Komponente kommuniziert mit Sonic hinsichtlich des Konfigurations- und Ereignisvorgangs sowie hinsichtlich der Erstellung korrelierter Ereignisse. Wenn Sie eine geänderte Konfigurationsdatei verwenden, ist Dateizugriff erforderlich.

Collector Manager

Sentinel-Komponente	Sentinel-Anwendung	Sentinel-Service	Sentinel-Vorgang	Funktionszusammenfassung	Anschlusstyp	Erforderliche Berechtigungen	Erläuterung der Berechtigung
Sentinel Wizard/Collector Manager	-	Collector Manager	agentengine.exe	Der Collector Manager-Vorgang verwaltet Collector Engines (initiiert Collector Engine-Vorgänge), veröffentlicht Systemstatusmeldungen, führt die globale Filterung von Ereignissen durch und nimmt referenzielle Zuordnungen vor. Der Collector Engine-Vorgang führt ein Collector-Skript aus, mit dem unverarbeitete (rohe) Ereignisse von Sicherheitsgeräten und -systemen normalisiert werden.	HINWEIS: Je nach Verbindungstypen sind für Collector Manager andere Berechtigungen erforderlich.		
					Seriell – Daten werden von einem seriellen RS-232C Anschluss gelesen	Berechtigung für Lese-/Schreibvorgänge an einem seriellen Anschluss	Collector Engine-Berechtigung für Lese-/Schreibvorgänge an einem seriellen Anschluss
					Socket – eine TCP-Socket-Verbindung	Netzwerkzugriff – Lese-/Schreibvorgänge vom Netzwerk-Socket; Berechtigung zur Verbindungsinitiierung	Initiierung einer Verbindung mit einem Netzwerkendpunkt sowie Lese-/Schreibvorgänge für dieses Socket durch Collector Engine
					Datei neu – liest nur Sicherheitsereignisdaten, die einer Datei hinzugefügt werden, nachdem das Skript gestartet wurde (liest vom Ende der Datei)	Lese-/Schreibzugriff für Dateien	Collector Engine-Lesevorgang in der ersten angegebenen Datei; das Schreiben erfolgt in die zweite angegebene Datei
					Datei alle – liest sämtliche Sicherheitsereignisdaten in einer Datei	Lese-/Schreibzugriff für Dateien	Collector Engine-Lesevorgang in der ersten angegebenen Datei; das Schreiben erfolgt in die zweite angegebene Datei

Sentinel-Komponente	Sentinel-Anwendung	Sentinel-Service	Sentinel-Vorgang	Funktionszusammenfassung	Anschlusstyp	Erforderliche Berechtigungen	Erläuterung der Berechtigung
					Permanenter Vorgang – startet einen permanenten Vorgang, wenn der Anschluss aktiviert wird, sorgt durch Empfangs- und Übertragungsstatusangaben für die Kommunikation zwischen dem diesem Anschluss zugewiesenen Collector und einer externen Anwendung und wird für die aktive Dauer des Anschlusses fortgesetzt.	Berechtigung zum Ausführen des definierten permanenten Vorgangs. (Hinweis: Wenn Sie EventLog.exe als permanenten Vorgang zum Sammeln von NT-Protokollen über WMI verwenden, muss Collector Manager auf WMI zugreifen können)	Collector Engine-Ausführung des definierten Vorgangs gemäß der aktuellen Berechtigungsstufe
					Temporärer Vorgang – sorgt durch Empfangs- und Übertragungsstatusangaben für die Kommunikation zwischen dem diesem Anschluss zugewiesenen Collector und einer externen Anwendung. Temporäre Vorgänge können mehrmals gestartet werden.	Berechtigung zum Ausführen des definierten temporären Vorgangs	Collector Engine-Ausführung des definierten Vorgangs gemäß der aktuellen Berechtigungsstufe
					SNMP – empfängt SNMP v1-, v2- und v3-Traps	Netzwerkzugriff – Lese-/Schreibvorgänge vom Socket	Collector Manager sendet/empfängt SNMP-Traps
					Keine	n/z	n/z

Sentinel-Komponente	Sentinel-Anwendung	Sentinel-Service	Sentinel-Vorgang	Funktionszusammenfassung	Anschlusstyp	Erforderliche Berechtigungen	Erläuterung der Berechtigung
Sentinel Wizard/Collector Builder	Collector Builder	-	agentbuilder.exe	Eine grafische Benutzeroberfläche (Graphical User Interface, GUI), über die Collectors erstellt, konfiguriert und gesteuert werden können. Über die GUI können lokale Collectors ausgeführt bzw. Collectors auf Wizard-Remote-Systemen gesteuert werden.	Lese-/Schreibzugriff für Dateien		Collector Builder-Lese-/Schreibvorgänge von Collector-Skripts in %WORKBENCH_HOME%/Elements
					Lese-/Schreibzugriff für Dateien		Collector Builder-Lese-/Schreibvorgänge der Anschlusskonfigurationsdatei in %WORKBENCH_HOME%/Agents
					Lese-/Schreibzugriff für Dateien		Collector Builder-Zugriff auf %ESEC_HOME%/.uuid
					Netzwerkzugriff – Lese-/Schreibvorgänge vom Netzwerk-Socket; Berechtigung zur Verbindungsinitiierung		Collector Builder-Herauf-/Herunterladevorgang von Collectors und erhaltenen Collector Manager-Zustandsmeldungen

Sentinel Communication

Sentinel-Komponente	Sentinel-Anwendung	Sentinel-Service	Sentinel-Vorgang	Funktionszusammenfassung	Erforderliche Berechtigungen	Erläuterung der Berechtigung
iSCALE/MOM	SonicMQ	Sentinel Kommunikation	sonicmf.exe	Unter Windows ist Sentinel Communication ein Service und trägt die Bezeichnung iSCALE – Message Oriented Middleware (MOM). Die iSCALE-Komponente stellt ein Java Message Service-(JMS-)Framework für die Kommunikation zwischen Vorgängen zur Verfügung. Prozesse können über einen Broker kommunizieren. Dieser ist für die Weiterleitung und Pufferung der Nachrichten verantwortlich. Es können mehrere Broker kommunizieren, z. B. zum Überwinden von Firewalls oder zum Lastausgleich. Die Sentinel-Vorgänge kommunizieren über einen Herausgeber-/Abonnent-Mechanismus miteinander. Hierdurch kann ein Vorgang eine Meldung in einem Themenkanal veröffentlichen, der von mehreren Abonnenten genutzt wird, ohne dass dem Veröffentlichungsvorgang bekannt ist, welcher Vorgang ihn abonniert. Abonnenten können veröffentlichte Meldungen von Herausgebern empfangen, ohne zu wissen, welche Herausgeber verfügbar sind. Hierdurch verringert sich der Konfigurationsaufwand und die Systemstabilität und -skalierbarkeit werden erhöht. Wenn beispielsweise ein neuer Wizard dem System hinzugefügt wird, ist auf Sentinel-Seite keinerlei Konfiguration erforderlich. Der Herausgebervorgang veröffentlicht Meldungen über Themen (Kanal) und der Abonnentvorgang abonniert Themen. Anschließend leitet der Message Broker Meldungen von Herausgebern an Abonnenten weiter, basierend auf den Themen, für die sie sich registriert haben.	Berechtigungen für den Zugriff auf die eigene eingebettete Datenbank, das eigene Installationsverzeichnis (%ESEC_HOME%\3rdparty\SonicMQ) und eigene Dateien	Sonic greift auf die eigene eingebettete Datenbank, auf das eigene Installationsverzeichnis (%ESEC_HOME%\3rdparty\SonicMQ) und eigene Dateien zu.

Datenbankserver (ohne DAS)

Sentinel-Komponente	Sentinel-Anwendung	Sentinel-Service	Sentinel-Vorgang	Funktionszusammenfassung	Erforderliche Berechtigungen	Erläuterung der Berechtigung
-	-	-	-	Sentinel-Datenbank einrichten	-	Der ODBC-(Open Database Connectivity-)Treiber bzw. Oracle-Treiber muss auf die Sentinel-DB verweisen.

Datenbankserver (mit DAS)

Eine Zusammenfassung/Aufschlüsselung der Sentinel-Datenbankzugriffsberechtigungen sowie die entsprechenden Details finden Sie in folgender Dokumentation:
Anhang A – Benutzer, Funktionen und Zugriffsberechtigungen der Sentinel-Datenbank

Sentinel-Komponente	Sentinel-Anwendung	Sentinel-Service	Sentinel-Vorgang	Funktionszusammenfassung	Erforderliche Berechtigungen	Erläuterung der Berechtigung
-	-	-	-	Sentinel-Datenbank einrichten	-	Der ODBC-(Open Database Connectivity-)Treiber bzw. Oracle-Treiber muss auf die Sentinel-DB verweisen.
Sentinel Server	-	Sentinel/Watch Dog.exe	das_binary	Einfügevorgänge für Ereignisse und korrelierte Ereignisse	Netzwerkzugriff; Zugriff auf DB für ESEC-Instanz als ESECAPP erforderlich	Die Komponente kommuniziert mit Sonic. Sie kommuniziert hinsichtlich des Datenabrufs über JDBC mit der DB und hinsichtlich der Ereignisseinfügung mit ADO (ActiveX Data Object), wenn die ADO-Ladestrategie den Setup-Wert aufweist.
			das_query	wird für alle anderen Datenbankoperationen verwendet	Netzwerkzugriff; Zugriff auf DB für ESEC-Instanz als ESECAPP erforderlich; Berechtigung zur Vorgangsausführung erforderlich	Die Komponente kommuniziert mit Sonic. Sie kommuniziert hinsichtlich des Datenabrufs über JDBC mit der DB.
			activity_container	Ausführen und Konfigurieren des Aktivitätsservices	Netzwerkzugriff; Zugriff auf DB für ESEC-Instanz als ESECAPP erforderlich; Berechtigung zur Vorgangsausführung erforderlich	Die Komponente kommuniziert mit Sonic. Sie kommuniziert hinsichtlich des Datenabrufs und der Dateneinfügung über JDBC mit der DB.
			workflow_container	Konfigurieren des Workflow-Services (iTRAC)	Netzwerkzugriff; Zugriff auf DB für ESEC_WF-Instanz als ESECAPP erforderlich; Berechtigung zur Vorgangsausführung erforderlich	Die Komponente kommuniziert mit Sonic. Sie kommuniziert hinsichtlich des Datenabrufs und der Dateneinfügung über JDBC mit der DB.

Eine Zusammenfassung/Aufschlüsselung der Sentinel-Datenbankzugriffsberechtigungen sowie die entsprechenden Details finden Sie in folgender Dokumentation:
Anhang A – Benutzer, Funktionen und Zugriffsberechtigungen der Sentinel-Datenbank

Sentinel-Komponente	Sentinel-Anwendung	Sentinel-Service	Sentinel-Vorgang	Funktionszusammenfassung	Erforderliche Berechtigungen	Erläuterung der Berechtigung
			das_rt	Konfigurieren der Active Views-Funktion in der Sentinel-Steuerungskonsole	Netzwerkzugriff; Zugriff auf DB für ESEC-Instanz als ESECAPP erforderlich	Die Komponente kommuniziert mit Sonic. Sie kommuniziert hinsichtlich des Datenabrufs über JDBC mit der DB.

Reporting Server

Sentinel-Komponente	Sentinel-Anwendung	Sentinel-Service	Sentinel-Vorgang	Funktionszusammenfassung	Erforderliche Berechtigungen	Erläuterung der Berechtigung
-	-	-	-	Crystal Reports XI bzw. Crystal Enterprise 9 Standard ist eines der Werkzeuge zur Berichterstellung in Sentinel.	-	Der ODBC-(Open Database Connectivity-) Treiber bzw. Oracle-Treiber muss auf die Sentinel-DB verweisen.

Glossar

HINWEIS: Die Begriffe „Agent“ und „Collector“ sind austauschbar. Im Folgenden werden Agenten als „Collectors“ bezeichnet.

Abfragemanager-Prozess (query_manager)	Der Abfragemanager (query_manager) empfängt vom Sentinel Control Center Anfragen zu kurzen Abfragen und zum Anzeigen von Details und leitet diese über DAS an die Datenbank weiter. Mit den Anforderungen vom Sentinel Control Center werden die erforderlichen Ereignisse mithilfe von Kriterien oder Filtern definiert. Falls ein Filter verwendet wird, empfängt der Abfragemanager die Filterdefinition und konvertiert den Filter in ein XML-Kriterium. Der Abfragemanager sendet die Anforderung anschließend an die Datenbank. Es können nicht alle Filter vollständig in XML konvertiert werden. Falls der Filter vollständig konvertiert wurde, weist der Abfragemanager DAS an, die Antwort direkt an das Sentinel Control Center zu senden. Falls der Filter reguläre Ausdrücke enthält, die nicht in XML konvertiert werden können, konvertiert der Abfragemanager den entsprechenden Teil und generiert ein konservatives XML-Kriterium, das einen übergeordneten Satz der erforderlichen Ereignisse zurückgibt. In diesem Fall wird DAS vom Abfragemanager angewiesen, das Ergebnis an ihn zurückzugeben. Wenn die Antwort an den Abfragemanager zurückgegeben wird, wird sie im Arbeitsspeicher gefiltert und die Ereignisse, die den Filter passieren, werden an das Sentinel Control Center gesendet.
Advisor	Ein integriertes System mit SecurityNexus-Datenbank für Anfälligkeiten, das Querverweise zwischen Echtzeit-Ereignissen und bekannten Anfälligkeiten bereitstellt.
Agent	Siehe Collector
Agent Builder	Siehe Collector Builder
Agent Engine	Siehe Collector Engine
Agenten-Manager	Collector Manager

Aggregation und Ereignisnormalisierung

Als Aggregation wird der Vorgang bezeichnet, bei dem individuelle Datenelemente mit geringer Bedeutung verwendet und kombiniert werden und so ein Datenelement erstellt wird, das möglicherweise von hoher Relevanz ist. Die einzelnen Teile eines Ereignisses, z. B. der Ereignisname, das Ereignisdatum, Quellen-IP, Ziel-IP, UUID, Sensortyp usw., besitzen einzeln betrachtet keine solche Bedeutung. Werden diese Elemente jedoch zusammengefügt zu einem Ereignis, könnte es sich bei diesem Ereignis möglicherweise um ein wichtiges Ereignis handeln, nämlich einen Angriff auf das Netzwerk, mit dem der Bestand ausgenutzt wird. Das Speichern eines vollständigen Ereignisses führt dazu, dass doppelte Informationen gespeichert werden. Dies ist beispielsweise der Fall, wenn in einem nichtaggregierten System für zehn Ereignisse, die weitgehend identisch sind, mit Ausnahme des jeweiligen Ereignisdatums, alle einzelnen Ereigniselemente, d. h. somit zehnmal identische Datenelemente (Ereignisname, Sensortyp usw.), gespeichert werden. Durch die Aggregation werden identische Datenelemente nur einmal gespeichert und bleiben dann eine Stunde als ausgeführtes Konto bestehen.

Ereignisdaten werden umgewandelt, zusammengefasst und in Zusammenfassungstabellen gespeichert. Anschließend können Zusammenfassungsberichte für vorberechnete Zusammenfassungen ausgeführt werden, sodass in Echtzeit-Ereignistabellen weniger Abfragen enthalten sind. Die Ereignis-Aggregations-Engine erfasst binäre Ereignisdaten, wandelt diese in eine normalisierte Ereignisstruktur um und erstellt auf der Grundlage eines vordefinierten Satzes von Zusammenfassungsdefinitionen eine Zusammenfassung. Mit der Ereignis-Aggregations-Engine werden Ereignisse fast in Echtzeit verarbeitet; es entsteht nur ein minimaler Overhead im Vergleich zum Echtzeit-Sentinel-System.

Analyse

Im Sentinel Control Center; ermöglicht Verlaufsberichte. Verlaufs- und Anfälligkeitsberichte werden auf dem Crystal[®]-Webserver veröffentlicht. Diese werden direkt mit der Datenbank ausgeführt und im Sentinel Control Center auf den Registerkarten „Analyse“ und „Advisor“ in der Navigationsleiste angezeigt.

Anfälligkeitsvisualisierung

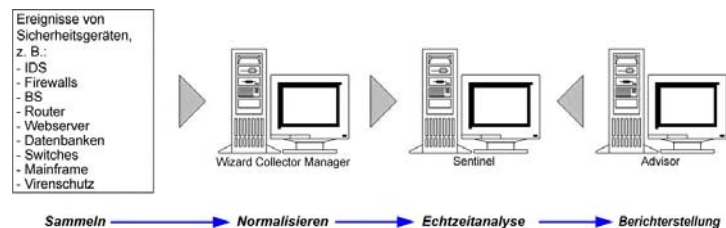
Eine grafische Darstellung von Echtzeitdaten in anfälligen Systemen; ist verfügbar für ein Ereignis als aktuelle Anfälligkeit oder Ereigniszeitanfälligkeit.

Bestandsverwaltung

Die Bestandsverwaltung wird dazu verwendet, ein Ereignis oder mehrere Ereignisse mit Inventar- und Anfälligkeitsinformationen zu verknüpfen, um damit den Bestand der Organisation wirksam schützen zu können. Es wird zwischen zwei Bestandstypen unterschieden, physischem Bestand und immateriellem Bestand. Als physischer Bestand zählt die Hardware, als immaterieller Bestand gelten Services und Anwendungen.

Collector

Als Collector wird ein Rezeptor bezeichnet, der unverarbeitete (rohe) Ereignisse aus Sicherheitsgeräten und Programmen sammelt und normalisiert. Diese Ereignisse können korreliert, gemeldet und für Antworten auf Vorfälle verwendet werden.



Es gibt drei Collector-Stufen:

- Unterstützte Collectors (T1)
- Dokumentierte Collectors (T2)
- Beispielcollectors (T3)

Collectors bestehen aus:

- Schablonendateien
- Parameterdateien
- Suchdateien
- Zuordnungsdateien

Collector Builder

Eine grafische Benutzeroberfläche (GUI), in der auf Grundlage der Collectors Regeln erstellt werden können, mit denen Daten aus unterschiedlichen Quellen gesammelt, gefiltert und normalisiert werden können sowie relevante Informationen sicher an Sentinel Server übermittelt werden, was zum Überwachen des Datenverkehrs genutzt werden kann.

Collector Engine

Verarbeitet die Schablonenlogik für die einzelnen Ports. Eine Collector Engine führt einen entsprechenden Port aus.

Collector Manager	Wizard-Backend-Komponente, die Collectors und Meldungen über den Systemstatus verwaltet.
CorrelatedEventUUID	Die Ereigniskennung des korrelierten Ereignisses, die von der ausgelösten Regel generiert wurde.
Correlation Engine	Die Correlation Engine führt Analysen der eingehenden Ereignisse aus, um wichtige Muster zu ermitteln und korrelierte Ereignisse zu untersuchen, um Details zu ermitteln, die das Ausführen einer Regel bewirkt haben.
Correlation Engine-Prozess (correlation_engine)	Beim Correlation Engine-Prozess (correlation_engine) werden Ereignisse vom Wizard Collector Manager empfangen und korrelierte Ereignisse auf der Grundlage benutzerdefinierter Korrelationsregeln veröffentlicht.
das_aggregation.xml	Wird für die Aggregationsoperation verwendet.
das_binary.xml	Wird für Ereignisse und Einfügeoperationen für korrelierte Ereignisse verwendet.
das_itrac.xml	Wird zum Ausführen und Konfigurieren des Aktivitätsservice und zum Konfigurieren des Workflowservice verwendet.
das_query.xml	Gibt die Konfigurationsparameter für den Data Access Service (DAS) an, eine Komponente der Sentinel-Datenbank.
das_rt.xml	Gibt die Konfiguration der Active Views-Funktion in der Sentinel-Steuerungskonsole an.

Data Access Service-Prozess (DAS)

Der Data Access Service (DAS)-Prozess bildet den Persistenz-Service des Sentinel Servers und stellt eine Nachrichtenbus-Schnittstelle (iSCALE) für die Datenbank bereit. Er ermöglicht einen datengetriebenen Zugriff auf die Backend-Datenbank. Es wird eine XML-Anforderung von verschiedenen Sentinel Prozessen empfangen. Diese werden in eine Abfrage an die Datenbank umgewandelt. Die Ergebnisse der Datenbank werden verarbeitet und zurück in eine XML-Antwort konvertiert. Mit diesem Prozess werden Anforderungen unterstützt, um Ereignisse für Quick Query und Event Drill Down abzurufen, die Anfälligkeit von Informationen und Advisor-Informationen abzufragen und Konfigurationsinformationen zu manipulieren. Mithilfe von DAS erfolgt außerdem die Protokollierung aller vom Wizard Collector Manager empfangenen Ereignisse und Anforderungen zum Abrufen und Speichern von Konfigurationsinformationen.

Daten-Controller

Siehe Datensynchronisierungsprozess.

Datensynchronisierungsprozess (Daten-Controller)

Beim Datensynchronisierungsprozess (data_synchronizer) wird die Änderung von Konfigurationsdaten durch mehrere Benutzer verwaltet. Wenn ein Benutzer Daten über das Sentinel Control Center ändern möchte, wird dieser Datensatz vom data_synchronizer gesperrt. Detaillierte Informationen, von wem die Daten gesperrt wurden, werden für die anderen aktiven Sentinel Control Centers veröffentlicht und diese Daten können von keinem anderen Benutzer geändert werden. Falls ein Sentinel Control Center geschlossen wird, bevor zuvor gesperrte Daten wieder freigegeben werden, erfolgt eine Zeitüberschreitung für die Sperre.

Ereignis

Ein Ereignis ist eine Aktion oder ein Vorgang, die bzw. der von einem Sicherheitsgerät (externes Ereignis) oder einem Prozess (intern) ermittelt wird. Ereignisse können sich auf die Sicherheit, die Leistung oder auf Informationen beziehen. Bei einem externen Ereignis kann es sich beispielsweise um einen Angriff handeln, der von einem Intrusion Detection System (IDS) ermittelt wird, um eine erfolgreiche Anmeldung, die von einem Betriebssystem erkannt wird, oder um einen benutzerdefinierten Vorgang, wie etwa den Zugriff auf eine Datei durch einen Benutzer. Informationsbezogene Ereignisse sind interne Ereignisse. Interne Ereignisse geben eine Änderung im Status eines Prozesses an. Beispielsweise das Anhalten eines Ports.

Ereignis-Echtzeit	Die Möglichkeit, Ereignisse zu überwachen, wenn sie eintreten, und Abfragen an diese Ereignisse zu senden. Die Ereignisse können in einem Tabellenformat oder in einer grafischen 3-D-Darstellung überwacht werden.
Ereignis-ID-Nummer	Eine Zahl, die einem Ereignis zugeordnet ist.
Ereigniskonfiguration	<p>Durch die Ereigniskonfiguration (Teil des Zuordnungsservices) können Sie Folgendes ausführen:</p> <ul style="list-style-type: none"> ▪ Überwachung der Regelkonformität aktivieren ▪ Richtlinienkonformität aktivieren ▪ Prioritätenfestlegung für Antworten aktivieren ▪ Sicherheitsdaten als analysierte bezogene Geschäftsoperationen aktivieren ▪ Verantwortlichkeit verbessern <p>Als Ereigniskonfiguration wird das Zuweisen von Namen zu bestehenden Labels bezeichnet. Beispielsweise das Umbenennen von „Ct2“ zu „City“. Die Änderungen werden in Filtern und Korrelationsregeln fortgesetzt.</p>
Ereignisnormalisierung	Siehe Aggregation
Ereignisrouter	Mit dem Ereignisrouter erfolgt die Ereigniszuordnungstransformation und -filterung.
Erweiterte Korrelationsregel	<p>Ermöglicht das Erstellen einer Korrelationsregel, die alle Funktionen einer einfachen Korrelationsregel umfasst. Außerdem wird ein Ereignis gesendet, wenn ein Ereignissatz unterschiedliche META-Tag-Werte enthält, z. B. Sensoren die sich innerhalb oder außerhalb der Firewall befinden. Beispielsweise kann mit einer erweiterten Korrelationsregel nach Ereignissen gesucht werden, die von derselben Quellen-IP-Adresse zu derselben Ziel-IP-Adresse erfolgten, denselben Ereignisnamen aufweisen und innerhalb und außerhalb der Firewall aufgetreten sind. (Dies bedeutet, dass möglicherweise ein Angriff durch die Firewall gelangen konnte.)</p>
Exploit-Erkennung	Siehe Zuordnungsservice.

Filter

Mithilfe der Sentinel-Filter können Daten auf der Grundlage spezifischer Kriterien verarbeitet werden. Dies gilt für im System eingehende Ereignisse sowie für Benutzer des Systems. Es gibt unterschiedliche Filterstufen:

- Collector – erfolgt über das Skript, mithilfe von Collector Builder
- globaler Filter – wird für alle Ereignisse angewendet, die von Wizards im System generiert wurden. Nur Ereignisse, die globale Filter passieren, werden an alle Sentinel-Prozesse gesendet.
- Sicherheitsfilter – wird auf aktive Benutzer angewendet. Mit diesem Filter werden die Ereignisse beschränkt, die ein aktiver Benutzer beobachten kann. Diese Filter werden vom Administrator zugewiesen.
- Anzeigefilter – wird für die Ansichtenschnittstelle angewendet. Mithilfe dieser Filter können Benutzer ihre Ereignisfenster für Echtzeitanalysen definieren. Diese Filter werden von den einzelnen Benutzern angewendet.

Es gibt zwei Filtertypen:

- öffentlich – Öffentliche Filter sind systemeigen. Öffentliche Filter können als Sicherheits- oder als Anzeigefilter verwendet werden. Sicherheitsfilter richten sich nach den Benutzerberechtigungen. Anzeigefilter bestimmen, welche Ereignisse in den Echtzeit-Tabellen, Grafiken und Diagrammen dargestellt werden.
- privat – Private Filter sind im Besitz des jeweiligen Benutzers. Private Filter sind Anzeigefilter. Sie können diese Filter freigeben, wenn Sie über die Berechtigung zum Anzeigen privater Filter verfügen.

Filter-Engine

Siehe Ereignisleistungs-Prozess.

Filter-Engine

Siehe Ereignisleistungs-Prozess.

Grundlegende Korrelationsregel

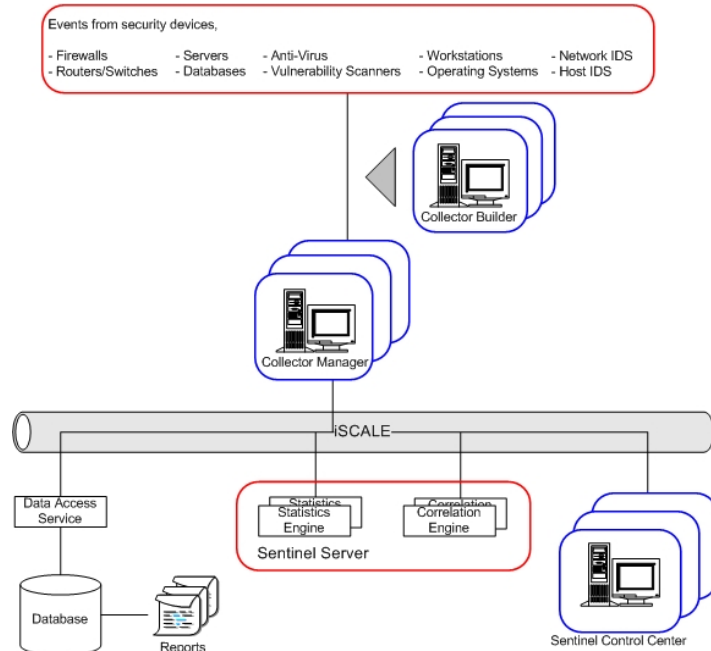
Sie können beliebige META-Tags zum Erstellen einer Korrelationsregel auswählen, mit der Sie zählen können, wie häufig bestimmte Bedingungen innerhalb eines bestimmten Zeitrahmens erfüllt werden. Beispielsweise kann mit einer grundlegenden Korrelationsregel nach derselben Quellen-IP-Adresse gesucht werden, die innerhalb von 5 Minuten fünfmal gemeldet wurde, auch wenn die Ereignisse über verschiedene Produkte (z. B. IDS (Intrusion Detection System) und Firewall) gemeldet werden.

Interne Ereignisse

iSCALE™

Siehe Systemereignisse.

Der Nachrichtenbus stellt ein JMS-Rahmenwerk (Java Message Service) für eine Zwischenprozess-Kommunikation bereit. Prozesse können über einen Broker kommunizieren. Dieser ist für die Weiterleitung und Pufferung der Nachrichten verantwortlich. Es können mehrere Broker kommunizieren, z. B. zum Überwinden von Firewalls oder zum Lastausgleich.



Die folgenden Prozesse kommunizieren über den Nachrichtenbus miteinander.

- Beobachtungsliste
- Ereignisleistung (Filter-Engine)
- Anzahl Ereignisse im Zeitraum (Statistik-Engine)
- Datensynchronisierung (Daten-Controller)
- Correlation Engine
- RuleLg Checker-Prozess (Korrelationsregel-Prüfung)
- Data Access Service-Prozess (DAS)
- Abfragemanager

iTRAC™

iTRAC umfasst die Automatisierung von Prozeduren und die Möglichkeit, auf Vorfälle zu reagieren. Sentinel stellt ein Workflow-Management-System bereit, das die Automatisierung von Prozeduren des SANS-Vorfallbehandlungsprozesses umfasst. Die wichtigsten Bestandteile von iTRAC sind:

- Worklist Handler – Anwendung, mit der von einer Aktivität zu einer anderen gewechselt werden kann.
- Activity Builder – Anwendung zum Erstellen eines eigenen benutzerdefinierten iTRAC
- Process Monitor – Überwacht die Aktivitäten (Schritte), die ausgeführt werden, um einen Prozess abzuschließen.

Korrelation

Der Vorgang des Analysierens von Sicherheitsereignissen zum Ermitteln von potenziellen Beziehungen zwischen zwei oder mehr Ereignissen. Durch die Korrelation werden schnelle Verknüpfungen von Prioritätsangriffen auf allgemeine Elemente der Ereignisdaten ermöglicht. Trends oder Muster bei Ereignissen niedrigerer Ebenen, die unterhalb von Sicherheitsschwellenwerten ausgeführt werden, können mithilfe der Korrelation effizienter identifiziert werden.

Sentinel umfasst fünf Korrelationsregeltypen. Dazu gehören:

- Beobachtungsliste
- Grundlegende Korrelation
- Erweiterte Korrelation
- FreeForm RuleLg

Korrelationsregel für Beobachtungsliste

Sie können eine Textzeichenkette angeben, mit der META-Tags in eingehenden Ereignissen von der Correlation Engine überprüft werden. Eine Regel kann beispielsweise nach einer spezifischen Quellen-IP-Adresse eines Hackers suchen und Sie jedes Mal benachrichtigen, wenn diese IP-Adresse in Ereignisnachrichten ermittelt wird.

Message Oriented Middleware (meldungsbasierte Middleware)

Siehe iSCALE™.

META-Daten

Als META-Daten werden Informationen zu Daten bzw. vordefinierte Variablennamen für META-Daten bezeichnet. Beispielsweise wird die Quellen-IP eines Angriffs im META-Tag „SourceIP“ gespeichert. Produktnamen werden im META-Tag „ProductName“ gespeichert. Daten zum Auffüllen von META-Tags werden entweder aus den Ereignisdaten extrahiert oder als Teil der Collector-Verarbeitung festgelegt.

META-Tag

In META-Tags werden META-Daten gespeichert.

MOM

Siehe iSCALE™.

Parameterdateien

Für Collectors sind Parameterdateien (PAR-Dateien) Tabellen, mit denen Parameternamen für die entsprechend ausgeführten Skriptdateien definiert werden. Sie werden verwendet, wenn eine Referenz im Analysecode vorliegt. Parameter bilden das Äquivalent zu Variablen. Parameter werden als Zeichenketten gespeichert. Zur Bearbeitung muss ein numerischer Wert in eine Zeichenkette konvertiert werden. Werden neue Werte für Parameter eingegeben, werden diese wirksam, nachdem Sie Ihr Skript erstellt haben. Beim Erstellen eines Skripts werden sie mit der Schablonendatei zusammengeführt.

Die Dateinamen der Ausführungsskriptdateien werden in der ersten Zeile der Tabelle angezeigt, die Parameternamen oder Labels werden in der ersten Spalte der Tabelle angezeigt. In der zweiten Zeile der Tabelle werden die Symbole definiert, die im Collector-Baum angezeigt werden. In den restlichen Zeilen werden die Variablen oder die Parameterwerte definiert, die für die Parameter verwendet werden, da diese zu einem bestimmten Skript gehören.

Werte innerhalb der Parameterdatei sind:

- META-Tags, Informationen und Kommentare – es sind mehr als 200 META-Tags verfügbar; 100 können vom Benutzer konfiguriert werden, die restlichen sind reserviert.
- Regel – Dateinamen werden in der Kopfzeile der Tabelle angezeigt, die Parameter selbst werden in der ersten Spalte der Tabelle angezeigt.
- Bitmap – zweite Zeile in der Tabelle. Es wird die Bitmap definiert, die für diese Datei verwendet wird. Die Bitmap wird in der Collectors-Liste angezeigt.

Parsing-Befehl

In Wizard; eine Skriptingschnittstelle auf hoher Ebene zur Bearbeitung von Daten. Als Analyse (Parsing) wird ein Prozess bezeichnet, bei dem ein Ereignis in seine Komponenten aufgeteilt wird.

Port	In Wizard; über Ports kann ein Collector die Sicherheitsereignisdaten im Netzwerk suchen, indem die IP-Adresse und andere Informationen zur Quelle (Sicherheitsgerät [Router, IDS, Switch usw.]) bereitgestellt werden. Mit jeder Zeile in der Portkonfigurations-Tabelle wird ein Collector-Skript für eine Ereignisquelle ausgeführt.
RuleLg Checker-Prozess (rulelg_checker)	Beim RuleLg Checker-Prozess (rulelg_checker) werden Filter- und Korrelationsregelausdrücke überprüft. Das Sentinel Control Center verwendet diese Ergebnisse, um zu ermitteln, ob ein Filter oder eine Korrelationsregel gespeichert werden kann.
Rx Buffer	Teil des Collector Manager; Standardgröße beträgt 50.000 Ereignisse. Der Empfangspuffer ist ein Parameter, der bearbeitet werden kann. Die Mindestgröße beträgt 5.000.
Rx Buffer Pointer	Der Receive Buffer Pointer (Empfangspufferzeiger) zeigt auf Datenbyte im Empfangspuffer. Vor den einzelnen ausgewerteten decide-Zeichenketten wird der Receive Buffer Pointer auf seinen verwalteten Wert zurückgesetzt (normalerweise 0).

Schablonendateien

Für Collectors; Schablonendateien können erstellt, bearbeitet und gelöscht werden und es können Zustände hinzugefügt werden. Mit Schablonen wird die Verarbeitung der Datensätze bestimmt. Die Entscheidungen beziehen sich mehrheitlich darauf, mit welchen Datensatztypen Sie arbeiten, sowie auf deren Format. Es gibt eine äquivalente Schablonendatei mit der Erweiterung .tem.

Schablonendateien basieren auf dem jeweiligen Zustand. Ein Zustand ist ein Entscheidungspunkt im logischen Fluss oder Pfad einer Schablone. Jeder Punkt (Zustand) enthält Informationen zur nächsten auszuführenden Verarbeitung. Zustände schließen Parameter ein, wenn die Schablone mit einer Parameterdatei zusammengeführt wird. Spezifische Werte ersetzen die Parameter. Wenn die Parameter durch spezifische Werte ersetzt werden, werden eine oder mehrere Skriptdateien erstellt.

Wenn ein Zustand in eine Schablone eingefügt wird, wird dem Zustand eine Zahl zugewiesen, die nicht geändert wird, unabhängig davon, an welche Position innerhalb der Schablone der Zustand verschoben wird.

Schnellabfrage

Siehe Abfragemanager.

Sentinel Control Center

Sentinel Control Center ist die zentrale Verwaltungskonsole, in der Zusammenfassungen und Verlaufsberichte angezeigt, Echtzeit-Ereignisse gefiltert und Vorfälle erstellt werden. Das Sentinel Control Center ermöglicht das Anzeigen von Ereignissen in Echtzeit, eine Systemübersicht zu Änderungen in Aktivitäten, die durch in Collectors festgelegte Einstellungen ausgelöst wurden, Verwaltung von Filtern, Berichterstellung, korrelierte Regeln und globale Filter und die Verwaltung von Sicherheitsereignissen durch Vorfälle.

Sentinel Server

Sentinel Server empfängt normalisierte Ereignisinformationen, die von Collectors im Wizard Collector Manager gesammelt wurden. Diese Ereignisse werden von Sentinel Server korreliert, um Muster zu finden und Bedrohungen und Berichte zu Echtzeitdaten und Verlaufsinformationen zu identifizieren, die im Sentinel Control Center angezeigt werden können.

Sequenzen (Start und Zurücksetzung)

Start- und Zurücksetzungssequenzen werden einem Port zugewiesen, der eine Reihe von Skripten ausführt, die verfügbar sind, wenn der Port gestartet oder angehalten wird. Ein Skript muss in eine Start- oder Zurücksetzungssequenz eingebunden sein, damit es vom Port verwendet werden kann. Ports aktivieren einen Collector, um im Netzwerk nach Wizard-Hosts zu suchen. Dazu wird die IP-Adresse oder ein Dateiname zum Host angegeben. Ports stellen für Sentinel außerdem Informationen zum Speicherort der Sensoren und des Collectors bereit, der zum Verwalten der Daten für diese Sensoren verwendet wird. Folgende Optionen können für Ports konfiguriert werden:

- Verbindungstyp
- Prozessname
- Socket-Informationen
- SNMP-Informationen
- Eingabe-/Ausgabedateinamen
- Collectorname

Skriptdatei

In Wizard; eine kompilierte Datei (*.asd), die aus der Collector-Schablonendatei, der Parameterdatei, der Suchdatei und der Zuordnungsdatei besteht.

Startsequenzen

Siehe Sequenzen.

Statistik-Engine

Siehe Prozess „Anzahl Ereignisse im Zeitraum“.

Suchdateien

Für Collectors stellen Suchdateien optionale Tabellen dar (LKP-Dateien), die zum Vergleich empfangener Werte verwendet werden können, um zu ermitteln, welche Aktionen bei Bedarf erforderlich sind, um auf Sicherheitsereignisse zu reagieren. In den Suchdateien sind Abgleichsklauseln enthalten, die zum Vergleich der einzelnen Zeichenketten verwendet werden. Auf der Grundlage der Abgleichsklauseln in einer spezifischen Suchdatei und den von den Sensoren empfangenen Daten wird mit dem LOOKUP-Befehl bestimmt, ob die gesuchte Zeichenkette gefunden wird oder nicht.

Optional können der Abgleichzeichenkette Parsing-Befehle zugeordnet werden. Die Parsing-Befehle werden ausgeführt, wenn eine Übereinstimmung gefunden wird.

Systemereignisse

Interne Ereignisse oder Systemereignisse sollen den Status oder Statusänderungen des Systems melden. Es werden zwei verschiedene Ereignistypen vom System generiert:

- Interne Ereignisse
- Leistungsereignisse

Interne Ereignisse haben Informationscharakter und beschreiben einen einzelnen Zustand oder eine Änderung des Systemzustands. Diese Ereignisse informieren darüber, wenn sich ein Benutzer anmeldet, ein Fehler bei der Authentifizierung eines Benutzers auftritt, ein Prozess gestartet oder eine korrelierte Regel gestartet wird. Leistungsereignisse werden regelmäßig generiert und beschreiben die durchschnittlich von unterschiedlichen Teilen des Systems verwendeten Ressourcen.

Unternehmensrelevanz

Siehe Zuordnungsservice.

Vorfälle

Zusammenfassen eines Satzes von Ereignissen, die von Interesse sein können (ähnliche Ereignisse oder einen Satz unterschiedlicher Ereignisse gruppieren, die ein Interessenmuster, z. B. einen Angriff, angeben).

Watchdog-Prozess

Watchdog ist ein Sentinel-Prozess, mit dem alle anderen Sentinel-Prozesse verwaltet werden. Falls ein anderer Prozess als Watchdog anhält, wird dieser Prozess von Watchdog erneut gestartet.

Wizard

Collector Builder und Collector Manager.

Wizard-Host

Eine Maschine, auf der die Collector Manager-Software installiert ist.

Workflow

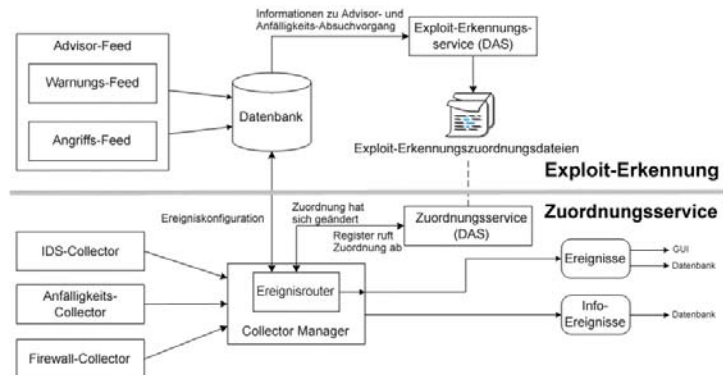
Siehe iTRAC™.

Zuordnungsdateien

Für Collectors sind Zuordnungsdateien optionale Dateien (.csv), mit denen eine schnelle Suche nach Schlüsseleinträgen möglich ist. Die CSV-Datei stellt einen relativen Pfad von einem Skriptverzeichnis des Collectors dar. Zurzeit können diese Dateien nicht im Collector Builder bearbeitet werden. Es ist jedoch eine Bearbeitung in Excel möglich.

Zuordnungsservice

Mit dem Zuordnungsservice von Sentinel kann eine sofortige, prozessfähige Benachrichtigung zu Angriffen auf anfällige Systeme erfolgen. Mit dem Zuordnungsservice wird eine Echtzeitverknüpfung zwischen den Ereignissen und den Ergebnissen von Anfälligkeitsprüfungen bereitgestellt. Damit werden Benutzer sofort automatisch benachrichtigt, wenn versucht wird, ein anfälliges System durch einen Angriff auszunutzen. Damit wird die Effizienz und Effektivität der Reaktion auf solche Vorfälle optimiert und eine verbesserte Verfügbarkeit kritischer Systeme und eine kosteneffiziente Sicherheit gewährt.



Zurücksetzungssequenzen

Siehe Sequenzen.

activity_container.xml	9-1
ALERT	3-4
APPEND	3-5
Auslöseoperations-Operator	
flow	7-25
intersection	7-25
union	7-25
Beispielkorrelationsregel	
Brute Force – Quelle und Ziel identisch ...	7-32
Denial of Service – Serviceunterbrechung	7-27
Fehler bei Anmeldung – von beliebiger	
Quelle in beliebigem Ziel	7-30
Fehler bei Anmeldung – von derselben	
Quelle bei demselben Ziel	7-31
Microsoft – Allgemeine Windows-	
Authentifizierung	7-35
Microsoft - IE	7-35
Microsoft – IIS	7-32
Microsoft – LM-Authentifizierung (LAN	
Manager)	7-35
Microsoft - MDAC	7-33
Microsoft - NETBIOS	7-34
Multiple Backdoor – einzelne Quelle	7-29
Multiple Backdoor – verschiedene Quellen	7-30
Pufferüberlauf – Serviceunterbrechung	7-27
Pufferüberlauf – von derselben Quelle zu	
demselben Ziel	7-31
Trojaner	7-29
UNIX - BIND/DNS	7-40
Unix – FTP	7-39
UNIX – Line Printer Daemon	7-40
UNIX – Remoteaufruf für Prozedur	7-37
UNIX – Remote-Services	7-39
UNIX – Secure Shell	7-38
UNIX – Sendmail-	7-40
Unix – SNMP	7-38
Unix – UNIX allgemein	7-41
Virenausbruch	7-28
Wurmausbruch	7-28
Beispielskorrelationsregel	
Microsoft – Anonyme Anmeldung	7-34
Microsoft – Remotezugriff auf	
Registrierung	7-36
Microsoft – SQL Server	7-33
Microsoft – Windows Scripting	7-36
UNIX – Apache-Webserver	7-37
Benutzer	
Standard	Siehe Standardbenutzer

Benutzerberechtigung	
Active Views	6-3
Advisor	6-5
Allgemein	6-2
Analyse	6-5
Benutzersitzungsverwaltung	6-7
Benutzerverwaltung	6-6
Collector-Verwaltung	6-4
DAS-Statistik	6-6
Ereignisdatei-Info	6-6
Globale Filter	6-6
Integrationsaktionen	6-2
iTRAC	6-3
iTRAC-Funktionsverwaltung	6-7
Korrelation	6-6
Menüelemente	6-3
Menükonfiguration	6-6
Öffentlicher Filter	6-2
Privater Filter	6-2
Schablonenverwaltung	6-3
Verwaltung	6-5
Vorfälle	6-4
Vorgangsverwaltung	6-4
Zusammenfassungsansichten	6-3
Beobachtungsliste	
Definition	7-5
Berechtigungen	
Collector Manager	D-2
Datenbankserver (mit DAS)	D-7
Datenbankserver (ohne DAS)	D-6
Reporting Server	D-9
Sentinel Communication	D-5
Sentinel Server	D-1
BITFIELD	3-8
BREAKPOINT	3-9
BYTEFIELD	3-10
CLEAR	3-12
CLEARTAGS	3-13
Collector Builder	4-1
Collector Engine	4-2
Collector Manager	4-1
Berechtigungen	D-2
COMMENT	3-14
COMPARE	3-14
ConnectionManager	9-2

CONSTANTTAGS.....	3-15	Hierarchie	2-2
CONVERT	3-16	Parameternamen.....	2-2
COPY	3-18	Regeln für den Receive Buffer Pointer	2-3
COPY-FROM-RX-BUFF	3-18	DECODE	3-29
COPY-FROM-RX-BUFF-UNTIL- SEARCH.....	3-18	DECODEMIME.....	3-29
COPY-FROM-STRING-TO-STRING- UNTIL-SEARCH	3-18	DELETE.....	3-30
COPY-STRING-TO-STRING	3-18	DispatchManager	9-2
CRC.....	3-20	DISPLAY	3-31
das_binary.xml	9-1	ELSE	3-32
erneut konfigurieren	9-2	Empfangspuffer	2-1
das_query.xml	9-1	ENCODE	3-33
erneut konfigurieren	9-2	ENCODEMIME.....	3-33
das_rt.xml.....	9-1	ENDFOR	3-34
DATE.....	3-21	ENDIF	3-34
Datenbankserver (mit DAS)		ENDWHILE.....	3-35
Berechtigungen	D-7	Erweiterte Korrelation	
Datenbankserver (ohne DAS)		Definition	7-6
Berechtigungen	D-6	Erweiterte Korrelationsregel	
Datentyp		Erstellen	7-14
array (Variablenarrays).....	2-7	ESEC_APP-Funktion	C-3
Derived Aggregate	2-7	ESEC_ETL-Funktion	C-9
fvar (Float-Variable).....	2-6	ESEC_USER-Funktion.....	C-12
ivar (Ganzzahlvariable)	2-6	esecadm	
number	2-6	Passwort ändern.....	10-1
quoted data	2-7	esecapp	
svar (Zeichenkettenvariable)	2-6	Passwort ändern.....	10-1
DATETIME	3-22	esecdba	
DBCLOSE	3-23	Passwort ändern.....	10-2
dbconfig.....	9-3	esecrpt	
DBDELETE	3-23	Passwort ändern.....	10-2
DBGETROW	3-24	EVENT.....	3-35
DBINSERT	3-25	Fensteroperations-Operator	
DBOPEN	3-26	flow	7-25
DBSELECT	3-27	intersection	7-25
DEC.....	3-28	union.....	7-25
decide-Zeichenketten		FILEA.....	3-38
Empfangspuffer	2-1	FILEL	3-39
Format	2-2	FILER	3-40

FILEW	3-41
Filteroperations-Operator	
flow	7-25
intersection	7-25
union	7-25
FOR	3-42
FreeForm RuleLg-Korrelation	
Definition	7-6
FreeForm RuleLg-Korrelationsregel	
Erstellen	7-19
GETCONFIG	3-43
GETENV	3-44
Grundlegende Korrelation	
Definition	7-6
Grundlegende Korrelationsregel	
Erstellen	7-10
Welche Ereignisse sollen im Musterabgleich ausgeschlossen werden	7-4
HEXTONUM	3-44
IF 3-45	
INC	3-47
INDICATOR	3-47
INFO_CLEAR_TAGS	3-48
INFO_CLOSE	3-48
INFO_CONSTANT_TAGS	3-49
INFO_CREATE	3-49
INFO_DUMP	3-50
INFO_PUSH	3-50
INFO_SEND	3-51
INFO_SETTAG	3-51
IPTONUM	3-55
Korrelation	
Ausgabe	7-49
Skriptparameter	7-49
Korrelationsbefehlszeile	8-1
affinityOneProcessor	8-3
configurationFile	8-2
dbRetries	8-2
dbTimeout	8-2
debug	8-1

help	8-3
inputChannel	8-1
logFile	8-3
logPeriod	8-3
mgmtInputChannel	8-2
mgmtOutputChannel	8-2
mgmtService	8-2
name	8-2
noStartupRules	8-2
outputChannel	8-1
outputExecuteChannel	8-2
outputUpdateChannel	8-1
ruleFile	8-1
service	8-2
useEventTime	8-3
useNullOutput	8-3
version	8-3
xmlruleFile	8-1
Korrelationsregel-Ausgabestruktur	7-49
LENGTH	3-56
LENGTH-OPTION2	3-56
LOOKUP	3-57
META-Tag	
CorrelatedEventUids	5-2
Criticality	5-2
Ct*	5-2
CustomerVar*	5-2
DataContact	3-36, 5-5
DateTime	5-3
DestinationAssetCategory	5-6
DestinationAssetId	5-7
DestinationAssetMaintainer	5-7
DestinationAssetName	5-6
DestinationAssetOwner	5-7
DestinationAssetValue	5-6
DestinationBuilding	5-6
DestinationBusinessUnit	5-7
DestinationCity	5-7
DestinationCountry	5-7
DestinationCriticality	5-6
DestinationDepartment	5-7
DestinationDivision	5-7
DestinationEnvironmentIdentity	5-6
DestinationFunction	3-37, 5-5
DestinationHostName	5-3
DestinationIP	5-3
DestinationLineOfBusiness	5-7
DestinationMacAddress	5-6
DestinationNetworkIdentity	5-6
DestinationOperationalContext	3-37, 5-5
DestinationPort	5-3
DestinationRackNumber	5-6
DestinationRoom	5-6
DestinationSensitivity	5-6

DestinationState	5-7
DestinationThreatLevel	3-37, 5-5
DestinationUserContext	3-37, 5-5
DestinationUserName	5-3
DestinationZipCode	5-7
DeviceCategory	3-36, 5-4
DeviceName	5-4
eSecTaxonomyLevel1	3-37, 5-5
eSecTaxonomyLevel2	3-37, 5-5
eSecTaxonomyLevel3	3-37, 5-5
eSecTaxonomyLevel4	3-37, 5-5
EventContext	3-36, 5-4
EventID	5-3
EventName	5-3
EventTime	5-3
ExtendedInformation	5-3
File Name (FN)	5-3
Message	5-4
MSSPCustomerName	3-36, 5-5
NormalizedAttackName	5-4
ProductName	5-4
Protocol (Prot)	5-4
ReporterName	5-4
ReservedVar1-10	5-4
ReservedVar11-20	5-4
ReservedVar21-25	5-4
ReservedVar40-43	5-5
ReservedVar49	3-37, 5-5
ReservedVar54-100	5-7
ReservedVar54-55	5-5
Resource	5-7
Rt1	5-7
Rt2	5-7
Rt3	5-7
SensorName	5-7
SensorType	5-8
Severity	5-7
SourceAssetCategory	5-5
SourceAssetID	5-6
SourceAssetMaintainer	5-6
SourceAssetName	5-5
SourceAssetOwner	5-6
SourceAssetValue	5-5
SourceBuilding	5-5
SourceBusinessUnit	5-6
SourceCity	5-6
SourceCountry	5-6
SourceCriticality	5-5
SourceDepartment	5-6
SourceDivision	5-6
SourceEnvironmentIdentity	5-5
SourceFunction	3-36, 5-5
SourceHostName	5-7
SourceID	5-7
SourceIP	5-7
SourceLineOfBusiness	5-6
SourceMacAddress	5-5
SourceNetworkIdentity	5-5

SourceOperationalContext	3-36, 5-5
SourcePort	5-8
SourceRackNumber	5-6
SourceRoom	5-6
SourceSensitivity	5-5
SourceState	5-6
SourceThreatLevel	3-36, 5-4
SourceUserContext	3-36, 5-5
SourceUserName	5-8
SourceZipCode	5-6
SubResource	5-8
VirusStatus	3-37, 5-5
Vulnerability	5-8
WizardAgent	5-8
WizardPort	5-8

NEGSEARCH.....3-59

Novell

Technischer Support	1-2
Website	1-2

NUMTOHEX.....3-60

NUMTOIP.....3-60

PARSER_ATTACHVARIABLE.....3-61

PARSER_CREATEBASIC.....3-63

PARSER_NEXT.....3-64

PARSER_PARSESTRING.....3-64

Parsing

Befehlsformat	3-3
---------------------	-----

Parsing-Befehl

Absuchen auf Anfälligkeit	3-3
ALERT	3-4
APPEND	3-5
Benachrichtigungsfunktion	3-1
BITFIELD	3-8
BREAKPOINT	3-9
BYTEFIELD	3-10
CLEAR	3-12
CLEARTAGS	3-13
COMMENT	3-14
COMPARE	3-14
CONSTANTTAGS	3-15
CONVERT	3-16
COPY	3-18
COPY-FROM-RX-BUFF	3-18
COPY-FROM-RX-BUFF-UNTIL-SEARCH	3-18
COPY-FROM-STRING-TO-STRING-UNTIL- SEARCH	3-18
COPY-STRING-TO-STRING	3-18
CRC	3-20
DATE	3-21
DATETIME	3-22

DBCLOSE	3-23	PARSER_PARSESTRING	3-64
DBDELETE	3-23	PAUSE	3-65
DBGETROW	3-24	POPUP	3-65
DBINSERT	3-25	PRINTF	3-66
DBOPEN	3-26	REGEXPREPLACE	3-68
DBSELECT	3-27	REGEXPSEARCH	3-70
DEC	3-28	REGEXPSEARCH_EXPLICIT	3-70
DECODE	3-29	REGEXPSEARCH_STRING	3-70
DECODEMIME	3-29	REPLACE	3-73
DELETE	3-30	RESET	3-74
Dienstprogrammfunktion	3-2	RXBUFFER	3-74
DISPLAY	3-31	SEARCH	3-75
ELSE	3-32	SET	3-76
ENCODE	3-33	SETBYTES	3-77
ENCODEMIME	3-33	SETCONFIG	3-78
ENDFOR	3-34	SHELL	3-79
ENDIF	3-34	SKIP	3-80
ENDWHILE	3-35	SKIPWORD	3-81
EVENT	3-35	SOCKETW	3-83
FILEA	3-38	STONUM	3-84
FILEL	3-39	STRIP	3-84
FILER	3-40	STRIP-ASCII-RANGE	3-84
FILEW	3-41	TBOSETCOMMAND	3-85
FOR	3-42	TBOSETREQUEST	3-88
Funktion für das Durchführen der		TIME	3-90
Fehlersuche	3-1	TOKENSIZE	3-90
Funktion für Dateiinteraktion	3-1	TOLOWER	3-92
Funktion für Datenbankinteraktion	3-1	TOUPPER	3-92
Funktion für logischen Vorgang	3-1	TRANSLATE	3-93
Funktion für Netzwerkinteraktion	3-1	TRIM	3-95
Funktion zur Bearbeitung von Rohdaten	3-2	WHILE	3-96
Funktion zur Bearbeitung von			
Zeichenketten	3-2	Parsing-Befehle	2-5
Funktion zur Verarbeitung von Variablen ...	3-3	Format	3-3
GETCONFIG	3-43	Verwendung von Arrays	3-3
GETENV	3-44	Parsing-Befehlsformate	3-3
HEXTONUM	3-44		
IF 3-45		PAUSE	3-65
INC	3-47	POPUP	3-65
INDICATOR	3-47	popup.cfg	4-2
INFO_CLEAR_TAGS	3-48	popup.exe	4-2
INFO_CLOSE	3-48	PRINTF	3-66
INFO_CONSTANT_TAGS	3-49		
INFO_CREATE	3-49	Regel Allgemeine Korrelation	
INFO_DUMP	3-50	Welche Ereignissesollenim	
INFO_PUSH	3-50	Musterabgleichenthaltensein	7-4
INFO_SEND	3-51	Regel für Beobachtungsliste	
INFO_SETTAG	3-51	Erstellen	7-6
IPTONUM	3-55	REGEXP_REPLACE	3-68
LENGTH	3-56	REGEXPSEARCH	3-70
LENGTH-OPTION2	3-56	REGEXPSEARCH_EXPLICIT	3-70
LOOKUP	3-57		
NEGSEARCH	3-59		
NUMTOHEX	3-60		
NUMTOIP	3-60		
PARSER_ATTACH_VARIABLE	3-61		
PARSER_CREATE_BASIC	3-63		
PARSER_NEXT	3-64		

REGEXPSEARCH_STRING.....	3-70	Sentinel Report-Benutzer	
Reguläre Ausdrücke.....	2-4	Passwort ändern.....	10-5
Sonderzeichen	2-4	Sentinel Server	
REPLACE.....	3-73	Berechtigungen	D-1
Reporting Server		Sentinel-Administrator	
Berechtigungen	D-9	Passwort ändern.....	10-3
Reservierte Ereignisvariable		Sentinel-Anwendungs-DB-Administrator	
s_P	3-35	Passwort ändern.....	10-4
Reservierte Ereignis-Variable		Sentinel-DB-Administrator	
i_Severity	3-35	Passwort ändern.....	10-3
s_BM	3-35	Serverfunktionen	C-14
s_CRIT	3-36	SET.....	3-76
s_CT1.....	3-36	SETBYTES.....	3-77
s_CT2.....	3-36	SETCONFIG.....	3-78
s_CT3.....	3-36	SHELL	3-79
s_CV1 – s_CV100.....	3-36	SKIP	3-80
s_DHN.....	3-36	SKIPWORD	3-81
s_DIP	3-36	Skriptparameter	
s_DP	3-35	%dt%	7-50
s_DUN.....	3-36	%sip%	7-50
s_EI	3-36	Skriptparameter	7-49
s_ET	3-35	%agent%	7-50
s_EVT	3-36	%all%	7-52
s_FN.....	3-36	%CorrelatedEventID%.....	7-50
s_PN	3-36	%crt%	7-50
s_Res	3-35	%ct1%	7-51
s_RN.....	3-36	%ct2%	7-51
s_RT1.....	3-36	%ct3%	7-51
s_RT2.....	3-36	%cv1% - %cv100%	7-51
s_RT3.....	3-36	%dhn%	7-51
s_RV1 – s_RV100.....	3-36, 3-37	%dip%	7-50
s_SHN.....	3-36	%dp%	7-51
s_SIP.....	3-36	%dun%	7-51
s_SN	3-36	%ei%	7-51
s_SP.....	3-35	%et%	7-50
s_ST.....	3-36	%evt%	7-50
s_SubRes.....	3-35	%fn%	7-51
s_SUN.....	3-36	%id%	7-50
s_VULN.....	3-36	%msg%	7-51
RESET	3-74	%pn%	7-51
RuleLg operator		%port%.....	7-50
and	7-21	%prot%.....	7-50
not	7-21	%res%	7-50
or	7-21	%rn%	7-51
RXBUFF	3-74	%rt1%.....	7-51
SEARCH	3-75	%rt2%.....	7-51
Sentinel Communication		%rt3%.....	7-51
Berechtigungen	D-5		

%RuleCount%	7-49	STONUM	3-84
%RuleDescription%	7-49	STRIP	3-84
%RuleDuration%	7-49	STRIP-ASCII-RANGE	3-84
%RuleLg%	7-49	TBOSSSETCOMMAND	3-85
%RuleName%	7-49	TBOSSSETREQUEST	3-88
%RulePattern%	7-50	TIME	3-90
%RuleResource%	7-49	TOKENSIZE	3-90
%RuleSeverity%	7-49	TOLOWER	3-92
%RuleSubResource%	7-49	TOUPPER	3-92
%RuleType%	7-49	TRANSLATE	3-93
%rv1% - %rv100%	7-51	TRIM	3-95
%sev%	7-50	Variablen	
%shn%	7-50	Sonderregeln	2-7
%sn%	7-50	WHILE	3-96
%sp%	7-50	Wizard	
%src%	7-50	Verzeichnisstruktur	4-3
%st%	7-50	Wizard-Dienstprogramme	
%sun%	7-51	Collector Builder	4-1
%vul%	7-50	Collector Engine	4-2
SOCKETW	3-83	Collector Manager	4-1
Standardbenutzer		popup.cfg	4-2
ESEC_CORR	6-1	popup.exe	4-2
esecadm	6-1	Wizard-Verzeichnisstruktur	4-3
esecapp	6-1	workflow_container.xml	9-1
esecdba	6-1		
esecrpt	6-1		
Standardbenutzerpasswort			
esecadm	10-1		
esecapp	10-1		
esecdba	10-2		
esecrpt	10-2		
Sentinel-Administrator	10-3		
Sentinel-Anwendungs-DB-Administrator..	10-4		
Sentinel-DB-Administrator	10-3		
Sentinel-Report-Benutzer	10-5		

