

FUJITSU Software openUTM Enterprise Edition (Unix-, Linux- und Windows-Systeme)
FUJITSU Software openUTM-CRYPT Enterprise Edition (Unix-, Linux- und Windows-Systeme)

Version 6.5A
April 2018

*1

Freigabemitteilung

Alle Rechte vorbehalten, insbesondere gewerbliche Schutzrechte.
Änderung von technischen Daten sowie Lieferbarkeit vorbehalten.
Haftung oder Garantie für Vollständigkeit, Aktualität und Richtigkeit der angegebenen Daten und Abbildungen ausgeschlossen. Wiedergegebene Bezeichnungen können Marken und/oder Urheberrechte sein, deren Benutzung durch Dritte für eigene Zwecke die Rechte der Inhaber verletzen kann.

© 2018 Fujitsu Technology Solutions GmbH

Die Marke Fujitsu und das Fujitsu Logo sind Marken oder registrierte Marken von Fujitsu Limited in Japan und in anderen Ländern. BS2000 ist eine Marke von Fujitsu Technology Solutions GmbH in Europa und in anderen Ländern.

1 Allgemeines	1
1.1 Bestellung	3
1.1.1 Nutzungsrechte	3
1.1.2 Nutzungsrechte für Verschlüsselung	3
1.2 Auslieferung	3
1.2.1 Software auf Datenträger	3
1.2.2 Software zum Download	4
1.2.3 Software für WS4UTM und BizXML2COBOL	4
1.3 Dokumentation	5
1.3.1 Verfügbarkeit	5
1.3.2 Dokumentation der Open Group-Schnittstellen	5
2 Software-Erweiterungen	6
2.1 Neue Funktionen in openUTM(Unix-, Linux- und Windows-Systeme) V6.5A6	
2.1.1 Lange Rechnernamen bis 64 Zeichen	6
2.1.2 Passwortlänge bis 16 Zeichen	7
2.1.3 Dead Letter Queue für LPAP und OSI-LPAP	7
2.1.4 Code-Konvertierungstabellen	8
2.1.5 Anwenderspezifische Fehlerbehandlung (Unix- und Linux-Systeme)	9
2.1.6 Zugangsdaten für den XA-Datenbank-Anschluss	9
2.1.7 KDCUPD bei geänderter Anzahl Datenbanken	9
2.1.8 Umhängen wartender Asynchron-Nachrichten für LTERM-, LPAP- und OSI-LPAP-Bündel	9
2.1.9 Sonstige Änderungen	10
2.2 Realisierte Change-Requests (CR)	11
3 Technische Hinweise	11
3.1 Ressourcenbedarf	11
3.2 SW-Konfigurationen	12
3.3 Produkt-Installation	14
3.3.1 Installation auf Unix-/Linux-Plattformen	14
3.3.2 Installation auf Windows-Plattformen	14
3.3.3 Installation zusammen mit C++ Umgebung auf Unix-/Linux-Plattformen	15
3.3.4 Installation zusammen mit openssl Umgebung auf Unix-/Linux-Plattformen	15
3.3.5 Installationsprotokoll	15
3.3.6 Hinweise: Kernelparameter für openUTM	15
3.4 Produkt-Einsatz	17
3.4.1 Hinweise zur Umstellung auf openUTM V6.5A	17
3.4.2 Umstellung einer stand-alone Anwendung in eine UTM-Cluster-Anwendung	17
3.4.3 Client-Programme mit lokaler Anbindung an openUTM	17
3.4.4 Binden von UTM OSI-TP Anwendungen auf Unix- und Linux-Plattformen	17
3.4.5 Binden und Starten von UTM-Anwendungen auf Unix- und Linux-Systemen	17
3.4.6 Binden von UTM-Anwendungen auf AIX-Systemen	18
3.4.7 openUTM V6.5A KDCDEF ohne Encryption	18
3.4.8 Hinweise zum Einsatz von openUTM-CRYPT V6.5A	18
3.4.9 Hinweise zur Beispielanwendung für UTM-Cluster-Anwendungen	18
3.5 Entfallene und gekündigte Funktionen	18
3.5.1 Entfallene Funktionen	18
3.5.2 Gekündigte Funktionen	19
3.6 Inkompatible Änderungen	19
3.7 Einschränkungen	21
3.7.1 Einschränkungen in UTM-Cluster-Anwendungen	21
3.7.2 Allgemeine Einschränkungen	21
3.8 Verhalten im Fehlerfall	22
4 Hardware-Unterstützung	23

1 Allgemeines

Der offene und universelle Transaktionsmonitor openUTM unterstützt die Erstellung und den Betrieb von Transaktionsanwendungen im FUJITSU Betriebssystem BS2000, in marktrelevanten Unix-Systemen (Solaris, HP-UX, AIX) sowie in Linux- und Windows-Systemen.

FUJITSU Software openUTM ermöglicht die Realisierung von Client-/Server-Architekturen sowie die Erstellung verteilter Transaktionsanwendungen in homogenen sowie auch heterogenen IT-Infrastrukturen (Multi-Tier-Architektur).

Im Produktumfang von openUTM sind bereits einige Tools kostenfrei enthalten:

- *1 • Die grafische Oberfläche WinAdmin zur Administration von UTM-Anwendungen wird auf der
*1 DVD mitgeliefert.
*1 Für openUTM WinAdmin gibt es eine eigene Freigabemitteilung.
- *1 • Das webbasierte Tool WebAdmin steht für die Administration von UTM-Anwendungen über
*1 einen Internet-Browser zur Verfügung.
*1 Für openUTM WebAdmin gibt es eine eigene Freigabemitteilung.
- Mit WS4UTM wird ein Tool zur Verfügung gestellt, welches auf komfortable Weise ermöglicht, Teilprogramme einer UTM-Anwendung als Web Service verfügbar zu machen. Aus einer in Java implementierten Web-Anwendung heraus kann ein UTM-Service als Webservice aufgerufen werden. Damit kann openUTM in SOA-Lösungen integriert werden (SOA = Service Oriented Architecture).
- Mit der Programmschnittstelle "XML für openUTM", kurz UTM-XML, besitzt openUTM eine komfortable Schnittstelle zum Erstellen und Bearbeiten von Daten in Form von XML-Dokumenten.
Für UTM-XML gibt es eine eigene Freigabemitteilung.
- Mit BizXML2Cobol ist es möglich, aus vorhandenen Service-Definitionen (WSDL-Beschreibung oder XML Schema) Cobol Datenstrukturen und Code zu erzeugen, der sich in vorhandene transaktionale Cobol-Applikationen integrieren lässt, so dass diese den vordefinierten Service realisieren. Damit wird der Top-Down-Ansatz (von der geschäftsrelevanten Definition zur Implementierung) in SOA Projekten auch für den Einbezug existierender Programmlogik unterstützt.

openUTM kann zusätzlich mit weiteren kostenpflichtigen Produkten kombiniert werden:

- Das Produkt FUJITSU Software openUTM-Client, welches Client-Server-Kommunikation mit openUTM-Server-Anwendungen ermöglicht, steht für das FUJITSU Betriebssystem BS2000 ebenso wie für Unix-, Linux- und Windows-Systeme als eigenes Produkt zur Verfügung.
Für openUTM-Client gibt es eine eigene Freigabemitteilung.
- Für die Kopplung mit CICS/IMS-Anwendungen auf IBM-Systemen über SNA LU6.2 gibt es das Zusatzprodukt FUJITSU Software openUTM LU62 (Unix-, Linux- und Windows-Systeme) V5.1.
Für openUTM LU62 gibt es eine eigene Freigabemitteilung.
- Für die Kopplung von UTM-Anwendungen mit Java EE Application Servern gibt es das Produkt FUJITSU Software BeanConnect. BeanConnect beinhaltet auch eine Komponente zur Erstellung von JAVA-Clients für openUTM.
Für BeanConnect gibt es eine eigene Freigabemitteilung.

- *1 Durch den zusätzlichen Einsatz der Verschlüsselungssoftware von openUTM und openUTM-Client mit
*1 Trägersystem UPIC ist ein verschlüsseltes Übertragen der Nachrichten zwischen der UTM-Anwendung
*1 und dem Client möglich.
*1 Zur Nutzung der Verschlüsselungsfunktionen ist ein zusätzlicher Produktbestandteil zu bestellen und zu
*1 installieren.

Weitere Informationen zu den oben aufgeführten Produkten sowie die Datenblätter finden Sie unter:
<http://de.ts.fujitsu.com/openUTM>

openUTM ist Teil des umfassenden Angebots der Middleware-Suite openSEAS.

Die Middleware-Suite openSEAS ermöglicht die einfache Integration einzelner Anwendungen und das Zusammenwirken von vorhandenen und neuen Anwendungen. openSEAS ergänzt sich perfekt mit der Fusion Middleware unseres strategischen Partners Oracle. Oracle Fusion Middleware ist eine marktführende Familie von vorintegrierten Produkten, die von Anwendungsinfrastruktur und Entwickleranwendungen bis zu Business Intelligence reicht.

Zusammen bietet dies die Plattform für:

- Individuelle geschäftskritische IT-Lösungen
- Unternehmensportale
- Integrationslösungen für Service Orientierte Architekturen
- Web-Anwendungen und Portale
- Geschäftsprozesse

*1 Diese Freigabemitteilung enthält in Kurzform die wesentlichen Erweiterungen, Abhängigkeiten und
*1 Betriebshinweise zu openUTM(Unix-, Linux- und Windows-Systeme) V6.5A und openUTM-
CRYPT(Unix-, Linux- und Windows-Systeme) V6.5A in den Betriebssystemen Solaris, HP-UX, AIX,
Linux und Windows sowie alle nötigen Informationen für den Umstieg von der Vorgängerversion
openUTM(Unix-, Linux- und Windows-Systeme) V6.4A.
Nachträglich bekannt gewordene Änderungen werden in dieser Datei aktualisiert und nachgeliefert.

*1 Der Inhalt entspricht dem Freigabestand: April 2018.

*1 Alle Änderungen gegenüber dem Freigabestand vom Dezember 2017 sind am linken Rand markiert.
*1 - Markierung *1 = Freigabestand V6.5A10 vom April 2018

Diese Freigabemitteilung ist, wie alle Freigabemitteilungen für Unix-, Linux- und Windows-Produkte,
über das Internet verfügbar, und zwar unter der URL:

<http://manuals.ts.fujitsu.com>

*1 Nachträglich bekannt gewordene Änderungen werden in dieser Datei aktualisiert und im Rahmen einer
*1 Korrekturfreigabe bekanntgegeben.

openUTM(Unix-, Linux- und Windows-Systeme) ist die Server-Komponente in dem Client/Server-Paket
openUTM Enterprise Edition (Unix-, Linux- und Windows-Systeme). In diesem Paket sind ferner
enthalten:

openUTM-Client (Client/Server-Kommunikation), openUTM WinAdmin und openUTM WebAdmin
(grafische Administrationsoberflächen).

Auf den verschiedenen Plattformen sind nicht immer alle Produkte verfügbar.

Das Trägersystem UPIC für UTM-Client-Anwendungen mit lokaler Anbindung (UPIC-Lokal) ist
Bestandteil von openUTM.

Für openUTM-Clients auf Unix-, Linux- und Windows-Plattformen, die remote über ein Netzwerk
angeschlossen sind, stehen zwei Trägersysteme (UPIC und OpenCPIC) mit unterschiedlicher
Mächtigkeit zur Auswahl.

Werden mit dem Einsatz der vorliegenden Produktversion eine oder mehrere Vorgängerversionen
übersprungen, so sind auch die Hinweise aus den Freigabemitteilungen (bzw. readme-Dateien) der
Vorgängerversionen zu berücksichtigen.

Die Wiedergabe von Namen, Warenbezeichnungen und dgl. in dieser Freigabemitteilung berechtigt
nicht zu der Annahme, dass diese Namen/Bezeichnungen ohne weiteres von jedermann benutzt
werden dürfen; oft handelt es sich um gesetzlich oder vertraglich geschützte Namen/Bezeichnungen,
auch wenn sie nicht als solche gekennzeichnet sind.

1.1 Bestellung

openUTM Enterprise Edition (Unix-, Linux- und Windows-Systeme) V6.5A kann über die für Sie zuständige Vertriebsgesellschaft bezogen werden.

Für openUTM Enterprise Edition (Unix-, Linux- und Windows-Systeme) V6.5A gelten die allgemeinen Bedingungen zum Kauf-, Wartungs-, Softwareüberlassungs- und Softwarebetreuungsvertrag.

1.1.1 Nutzungsrechte

Die Nutzungsrechte sind Bestandteil des Vertrages, den Sie beim Bezug von openUTM Enterprise Edition mit Fujitsu Technology Solutions abgeschlossen haben.

openUTM Enterprise Edition (Unix-, Linux- und Windows-Systeme) ist ein Lizenzprodukt. Für den Einsatz sind Nutzungsrechte notwendig. Sie werden als Basis-Nutzungsrecht pro System bzw. Partition plus Nutzungsrecht pro User angeboten.

Für Verteilte Transaktionsverarbeitung gibt es zusätzlich Nutzungsrechte in 5 verschiedenen Ausprägungen je nach Anzahl paralleler Verbindungen zwischen den Anwendungen.

Zur Entwicklung gibt es Entwicklungspakete mit dem Basisnutzungsrecht je System bzw. Partition und je 2 Usern für die Entwicklung von UTM-Anwendungen. Nutzungsrechte für weitere Entwickler können einzeln bezogen werden.

Die User-Nutzungsrechte von openUTM-Enterprise Edition beinhalten Client- und Server-User.

Nutzungsrechte für UTM-Cluster-Anwendungen:

Für die Nutzung der UTM Cluster Funktionalität gibt es keine spezielle Cluster Lizenz. Sie müssen openUTM auf jedem Knoten eigens lizenzieren, unabhängig davon ob Sie auf diesem Rechner eine stand-alone Anwendung oder eine Knoten-Anwendung betreiben.

*1 1.1.2 Nutzungsrechte für Verschlüsselung

*1 Die Verschlüsselungsfunktionen basieren auf der Funktionalität der openssl Bibliothek des
*1 verwendeten Systems. Hierzu sind keine zusätzlichen Nutzungsrechte nötig.

1.2 Auslieferung

1.2.1 Software auf Datenträger

openUTM

*1 Die Auslieferung der Dateien zu openUTM V6.5 erfolgt auf zwei verschiedenen DVD-Datenträgern:
- openUTM EED-DVD V6.5 (Software-DVD ohne Verschlüsselungsfunktion)
*1 - openUTM EED-DVD V6.5 Crypt (Software-DVD mit Verschlüsselungsfunktion)

*1 Auf diesen Datenträgern sind weitere Softwareprodukte der openUTM-Produktfamilie sowie für die Kommunikation gespeichert.

*1 Die openUTM EED-DVDs sind separat zu bestellen.

openUTM WinAdmin

Mit openUTM (Unix-, Linux- und Windows-Systeme) V6.5A wird auch das Produkt openUTM WinAdmin V6.5A ausgeliefert, mit dem Sie UTM-Anwendungen über eine komfortable grafische Oberfläche administrieren können.

1.2.2 Software zum Download

openUTM WebAdmin

Zusammen mit openUTM V6.5A wird zusätzlich das Produkt FUJITSU Software openUTM WebAdmin V6.5A freigegeben.

openUTM WebAdmin bietet eine weitere grafische Administrationsoberfläche, die im Unterschied zu WinAdmin auf einem Web-Server abläuft.

openUTM WebAdmin wird auf der openUTM Website <http://de.ts.fujitsu.com/openUTM> zum Download zur Verfügung gestellt.

XML für openUTM

Die Software "XML für openUTM" ist ein kostenloses Zusatzprodukt zu openUTM, für das Fujitsu Technology Solutions keine Fehlerbehebung übernimmt. Die Software ist beziehbar über

<http://de.ts.fujitsu.com/openutm>

Diese Komponente enthält den OpenSource XML-Parser libxml2, der zu den Bedingungen der MIT Lizenz bereitgestellt wird. Diese wird im Anwendungspaket ('rt') mit ausgeliefert und ist auch im Web unter

<http://www.opensource.org/licenses/mit-license.html>

zu finden.

1.2.3 Software für WS4UTM und BizXML2COBOL

Consulting Projektpakete, in deren Rahmen die Software „WebServices for openUTM“ (WS4UTM) bzw. BizXML2Cobol zur Verfügung gestellt wird, können über den für Sie zuständigen Sitz der Region von Fujitsu bezogen werden.

Die Handbücher können aus dem Web geladen werden:

<http://manuals.ts.fujitsu.com>

Die Nutzung der Software ist nur im Rahmen des im Projektpaket erworbenen Nutzungsrechtes gestattet.

1.3 Dokumentation

1.3.1 Verfügbarkeit

Die Dokumentation ist im Internet unter <http://manuals.ts.fujitsu.com> verfügbar.

1.3.2 Dokumentation der Open Group-Schnittstellen

<u>Titel:</u>	<u>Bestellnummer:</u>
CPI-C X/Open CAE Specification Distributed Transaction Processing: The CPI-C Specification, Version 2	ISBN 1 85912 135 7
Reference Model X/Open Guide Distributed Transaction Processing: Reference Model, Version2	ISBN 1 85912 019 9
TX X/Open CAE Specification Distributed Transaction Processing: The TX (Transaction Demarcation) Specification	ISBN 1 85912 094 6
XATMI X/Open CAE Specification Distributed Transaction Processing: The XATMI Specification	ISBN 1 85912 130 6

Die X/Open-Handbücher sind im Web verfügbar. Sie können gedruckt im Buchhandel unter der ISBN bezogen werden.

2 Software-Erweiterungen

2.1 Neue Funktionen in openUTM(Unix-, Linux- und Windows-Systeme) V6.5A

2.1.1 Lange Rechnernamen bis 64 Zeichen

OpenUTM unterstützt Rechnernamen, die bis zu 64 Zeichen lang sein können. Damit ist es nicht mehr notwendig, den Namen aus der UTM-Generierung mittels Konfigurations- oder Mapping-Dateien auf einen realen langen Namen abzubilden. Dazu wurden folgende Schnittstellen erweitert:

- Generierung
 - KDCDEF-Anweisungen CON, OSI-CON, PTERM und TPOOL:
Bei PRONAM= bzw. OSI-CON NETWORK-SELECTOR= kann ein bis zu 64 Zeichen langer Name für den Partner-Rechner angegeben werden.
 - KDCDEF-Anweisung CLUSTER-NODE:
Bei HOSTNAME= und VIRTUAL-HOST= kann ein bis zu 64 Zeichen langer Name angegeben werden.
 - KDCDEF-Anweisung MAX:
Bei HOSTNAME= kann ein bis zu 64 Zeichen langer Name angegeben werden.
- Programmierung
 - KDCS-Aufruf INFO SI:
 - Um die verfügbare Information vollständig zu erhalten, muss im Feld KCLA der Wert 180 angegeben werden.
 - Neue Felder KCHSTNML und KCPRONML für die Rückgabe langer Rechnernamen.
 - KDCS-Aufruf INIT PU:
 - Die Version der Schnittstelle wurde auf 6 erhöht
 - Um die verfügbare Information vollständig zu erhalten, muss im Feld KCLI der Wert 356 angegeben werden.
 - Neue Felder KCHSTNML und KCPRONML für die Rückgabe langer Rechnernamen.
 - KDCS-Aufruf SIGN ST:
 - Die Version der Schnittstelle wurde auf 4 erhöht
 - Um die verfügbare Information vollständig zu erhalten, muss im Feld KCLA der Wert 180 angegeben werden.
 - Der Knotenname im Feld KCCLNODE kann bis zu 64 Zeichen lang sein.
- Administrationsschnittstelle KDCADMI
 - Neue Struktur *kc_long_triple_str* im Identifikationsbereich für die Unterstützung langer Rechnernamen. Diese Struktur muss für alle Objekte außer KC_MUX verwendet werden. Daher wurde auch die Datenstrukturversion von KDCADMI geändert auf Version 10 (Feld *version_data* im Parameterbereich).
 - Neue Felder für die langen Rechnernamen:

Datenstruktur	Feldname(n)
<i>kc_cluster_node_str</i>	<i>hostname_long</i> und <i>virtual_host_long</i>
<i>kc_con_str</i> <i>kc_lses_str</i> <i>kc_lterm_str</i> <i>kc_pterm_str</i> <i>kc_tpool_str</i>	<i>proname_long</i>
<i>kc_osi_con_str</i>	<i>network_selector_long</i>
<i>kc_system_par_str</i>	<i>hostname_long</i>

- Kommando-Schnittstelle KDCADM
 - Kommando KDCINF:
Wenn ein Objekt vom Typ CON, LSES, LTERM, OSI-CON, PTERM oder TPOOL einen langen Rechnernamen enthält, dann wird die Information zu diesem Objekt in zwei Bildschirmzeilen ausgegeben.
 - Kommandos KDCLSES, KDCPOOL, KDCPTERM und KDCSWTCH:
Wenn ein Objekt einen langen Rechnernamen enthält, dann wird die Information zu diesem Objekt in zwei Bildschirmzeilen ausgegeben.
- Meldungen
Bei jeder Namensumsetzung zwischen langem Rechnernamen und kurzem lokalen Namen wird die Meldung K037 ausgegeben. Diese Meldung dient der Diagnose bei Verbindungsproblemen. Standardziel dieser Meldung ist die SYSLOG-Datei.
- Tool KDCNAMEINFO
Für die Zuordnung IP-Adresse zu Rechnernamen steht das Tool KDCNAMEINFO zur Verfügung.

2.1.2 Passwortlänge bis 16 Zeichen

Das Passwort einer UTM-Anwendung kann bis zu 16 Zeichen lang sein.

- Generierung
 - KDCDEF-Anweisung USER:
Bei PASS= kann ein bis zu 16 Zeichen langes Passwort angegeben werden.
Bei PROTECT-PW= wurde der Maximalwert für die Mindestlänge auf 16 erhöht.
- Programmierung
 - KDCS-Aufruf SIGN ON, CK:
Im Feld KCLA muss der Wert 16 angegeben werden.
 - KDCS-Aufruf SIGN CP:
Im Feld KCLA muss der Wert 32 angegeben werden.
 - KDCS-Aufruf SIGN ST:
 - Die Version der Schnittstelle wurde auf 4 erhöht.
 - Um die verfügbare Information vollständig zu erhalten, muss im Feld KCLA der Wert 120 angegeben werden.
 - Neues Feld KCRPASSL für die Rückgabe des Passworts aus dem UPIC-Protokoll.
- Administrationsschnittstelle KDCADMI
 - Datenstruktur *kc_user_str* und *kc_user_dyn2_str*:
Neue Felder *password16* und *protect_pw16_lth*.
- Kommando-Schnittstelle KDCADM
 - Kommando KDCUSER:
Bei PASS= kann ein bis zu 16 Zeichen langes Passwort angegeben werden.

2.1.3 Dead Letter Queue für LPAP und OSI-LPAP

Nachrichten an LU6.1- oder OSI-TP-Partner, die gelöscht werden, weil sie wegen eines permanenten Fehlers nicht zugestellt werden konnten, können jetzt in der Dead Letter Queue gesichert werden.

Dazu wurden die Schnittstellen wie folgt geändert:

- Generierung
KDCDEF-Anweisungen LPAP und OSI-LPAP:
Der neue Operand DEAD-LETTER-Q= steuert, ob nicht zustellbare Nachrichten, die aus ihrer Message Queue gelöscht werden, in der Dead Letter Queue gesichert werden sollen.

- Programmierung
Beim KDCS-Aufruf DADM mit KCOM=MV oder MA (Verschieben von Nachrichten aus der Dead Letter Queue) kann auch ein LPAP- oder OSI-LPAP-Partner angegeben werden.
- Administrationsschnittstelle KDCADMI
Datenstrukturen *kc_lpap_str* und *kc_osi_lpap_str*.
Neues Feld *dead_letter_q*, um die Sicherung nicht zustellbarer Nachrichten in der Dead Letter Queue zu steuern.

2.1.4 Code-Konvertierungstabellen

Für die Code-Konvertierung zwischen ASCII-kompatiblen Codes und EBCDIC-Codes stellt openUTM auf allen Plattformen vier Tabellen für Code-Konvertierungen bereit (zwei 8-Bit-Konvertierungen und zwei 7-Bit-Konvertierungen). Standardmäßig verwendet openUTM auf BS2000-, Unix- und Linux-Systemen eine 8-Bit-Konvertierung zwischen ISO8859-i und EBCDIC.DF.04-i und auf Windows-Systemen eine 8-Bit-Konvertierung zwischen Windows-1252 und EBCDIC.DF.04-F. Dazu wurden folgende Schnittstellen geändert:

- Generierung
 - KDCDEF-Anweisungen PTERM und TPOOL:
Die Operandenwerte MAP=SYS1, SYS2, SYS3 und SYS4 können auch auf Unix-, Linux- und Windows-Systemen angegeben werden. Die Angabe ist nur für Transportsystem-Anwendungen erlaubt (Partner-Typ APPLI oder SOCKET).
 - KDCDEF-Anweisungen OSI-CON und SESCHA:
Neue Operandenwerte MAP=SYS1, SYS2, SYS3 und SYS4 auf Unix-, Linux- und Windows-Systemen für die vier Konvertierungen.
- Programmierung
 - Die Sourcen mit den Konvertierungs-Tabellen werden auf allen Plattformen ausgeliefert und können auf allen Plattformen angepasst oder durch eigene Tabellen ersetzt werden.
- Administrationsschnittstelle KDCADMI
 - Datenstrukturen *kc_pterm_str* und *kc_tpool_str*.
Für das Feld map gibt es die zusätzlichen Werte '1', '2', '3' und '4' (für die vier Konvertierungen). Die Angabe ist nur für Transportsystem-Anwendungen erlaubt (Partner-Typ APPLI oder SOCKET).
 - Datenstrukturen *kc_lpap_str* und *kc_osi_con_str*.
Für das Feld map gibt es die zusätzlichen Werte '1', '2', '3' und '4' (für die vier Konvertierungen).

2.1.5 Anwenderspezifische Fehlerbehandlung (Unix- und Linux-Systeme)

Auf Unix- und Linux-Systemen kann eine eigene User Signal Routine erstellt werden. Diese Routine wird aufgerufen, sobald ein Signal aufgetreten ist. Sie ersetzt die Standard-Fehlerbehandlung von openUTM (abnormales Vorgangsende + PENDER-Dump), d.h. ein Vorgang kann damit trotz Fehler fortgesetzt werden.

Dazu wurden folgende Schnittstellen geändert:

- Programmierung
 - Neue Funktionen KCX_REG_SIGNAL_HANDLER und KCX_UN_REG_SIGNAL_HANDLER zum Registrieren und De-Registrieren der Signal Routine bei openUTM.
 - Neue Funktionen KCX_SET_RELOAD_FLAG zum Austausch eines Workprozesses nach Transaktionsende und KCX_WRITE_DUMP zum Erstellen eines UTM-Dumps ohne Vorgangsabbruch.

Die Funktionen stehen in COBOL und C zur Verfügung.

2.1.6 Zugangsdaten für den XA-Datenbank-Anschluss

- *1
- Die Zugangsdaten für den XA-Datenbank-Anschluss können optional (wieder) in den Startparametern angegeben werden.
 - Der Benutzername und das Passwort für den XA-Datenbank-Anschluss können per Administration (KDCADMI) geändert werden:
 - Operationscode KC_MODIFY_OBJECT:
 - Neuer Objekt-Typ KC_DB_INFO
 - Neue Werte KC_IMMEDIATE und KC_DELAY im Feld *subopcode1*, um den Zeitpunkt für die Passwortänderung zu steuern.
 - Datenstruktur *kc_db_info_str*:
Neue Felder *db_userid*, und *db_password*.
 - Die Zugangsdaten für den XA-Datenbank-Anschluss können bei KDCUPD übertragen werden:
 - TRANSFER-Anweisung: Neuer Operand DB-CREDENTIALS=.

2.1.7 KDCUPD bei geänderter Anzahl Datenbanken

KDCUPD überträgt alle Daten auch dann, wenn sich die Anzahl der Datenbanken in der neuen Generierung erhöht. Voraussetzung ist, dass sich die openUTM-Version nicht ändert und dass die Reihenfolge der bisherigen Datenbanken in der Generierung unverändert bleibt.

2.1.8 Umhängen wartender Asynchron-Nachrichten für LTERM-, LPAP- und OSI-LPAP-Bündel

Bei LTERM-, LPAP- und OSI-LPAP-Bündeln können wartende Asynchron-Nachrichten für Slave-LTERMS, Slave-LPAPs oder Slave-OSI-LPAPs, nach Ablauf der Wartezeit automatisch an einen Slave mit aufgebauter Verbindung umgehängt werden.

- Generierung, KDCDEF-Anweisung:
Neuer Operand MOVE-BUNDLE-MSGS=, um das automatische Umhängen zu steuern.
- Administrationsschnittstelle KDCADMI, Datenstruktur *kc_max_par_str*:
Neues Feld *move_bundle_msgs* für die Ausgabe des generierten Wertes.

2.1.9 Sonstige Änderungen

- Ausgabe der Generierungszeit an der Administrationsschnittstelle:
 - KDCADMI-Datenstruktur *kc_curr_par_str*.
Neue Felder *gen_date_xxx* und *gen_time_xxx* zur Ausgabe von Datum und Zeitpunkt der Generierung.
 - Kommando KDCINF STAT:
Zusätzliche Ausgabe von Datum und Zeitpunkt der Generierung.
- Meldung bei Verlust einer Nachricht an OSI-TP- und LU6.1-Partner
Wird eine Nachricht an einen OSI-TP- oder LU6.1-Partner gelöscht, weil sie wegen eines permanenten Fehlers nicht zugestellt werden konnte, dann wird die neue Meldung K239 ausgegeben.
- Verbesserte Performance bei der Kommunikation zwischen UPIC-Clients und UTM-Server-Anwendungen.
- Modifizieren von UTM-Meldungen
 - Meldungstexte von K- und P-Meldungen sind auch auf Windows modifizierbar.
Dazu werden das UTM-Tool KDCMMOD sowie das neue UTM-Tool KDCWMSG verwendet.
 - Meldungstexte und -ziele von U-Meldungen sind auf Unix-, Linux- und Windows-Systemen modifizierbar. Damit können U-Meldungen auch in die System-Protokolldatei SYSLOG ausgegeben werden.
Das bisher für das Ändern von K- und P-Meldungen beschriebene Verfahren kann jetzt auch für U-Meldungen angewendet werden.
- Der Portnummernbereich ist nicht mehr beschränkt, d.h. es können für Unix-, Linux- und Windows-Systeme alle Portnummern von 1 bis 65535 angegeben werden.
- Bei CON-Anweisungen sowie bei PTERM- und TPOOL-Anweisungen mit PTYPE=APPLI/UPIC-R/SOCKET ist die Angabe des Parameters PRONAM= jetzt Pflicht. Das gleiche gilt für den Parameter NETWORK-SELECTOR der OSI-CON-Anweisung.
- Das UTM-Tool KDCEVAL wurde wie folgt erweitert:
 - Neuer Operand SHOW-TSN im KDCEVAL-Kommando OPTION, um die wiederholte Ausgabe der TSN bzw. der PID in der Liste TRACE2 zu aktivieren oder zu deaktivieren.
 - Zusätzliche Ausgaben im Listenkopf wie z. B. Appli-Mode oder Test-Mode.
- *1 • Verschlüsselung
*1 Es wird die Verschlüsselungs-Funktionalität der openssl Bibliothek verwendet. Die openssl
*1 Bibliothek wird vom Anwender zur Verfügung gestellt.
- *1 • Installation auf Unix-/Linux-Systemen - Protokoll Dateien
*1 Sowohl im Installationsverzeichnis <install-path> als auch in den Verzeichnissen <install-
*1 path/32> bzw. <install-path/64> existieren die Protokolldateien utm.log bzw. upic.log.
*1 In der Protokolldatei steht nur dann „installation successfully“, wenn alle Aktionen der 32- bzw.
*1 64-Bit Installation erfolgreich waren.

- *1 • KDCEVAL auf Linux-, Unix- und Windows-Systemen
- *1 In UTM V6.5A10 gibt es folgende Erweiterungen zur Auswertung von UTM Anwendungen.
- *1 a) KDCEVAL - neuer Parameter für das Kommando OPTION
- *1 OPTION BEANCONNECT = YES | NO, bzw. OPTION BC = YES | NO
- *1 Diesen Parameter gibt es nur für KDCEVAL auf Linux-, Unix- und Windows-Systemen.
- *1 Bei BC = YES werden die Listen TASKS und TACLIST in dem gleichen Format
- *1 ausgegeben, das im BS2000 verwendet wird. D.h., es werden Informationen zu
- *1 Datenbanken (elapsed Zeiten sowie Anzahl Aufrufe) ausgegeben.
- *1 Da in nicht-BeanConnect-Anwendungen für DB-User-Calls keine Datenbank-Sätze
- *1 geschrieben werden, ist diese Art der Aufbereitung nur für BeanConnect Proxy
- *1 Anwendungen sinnvoll.
- *1 b) In den Listen TRACE und TRACE2 von KDCEVAL werden ab V6.5A10 auch
- *1 Datenbank-Sätze ausgegeben. Die Ausgabe der Datenbank-Sätze erfolgt immer, d.h.
- *1 unabhängig von der neuen Option BEANCONNECT=YES. Für nicht-BeanConnect-
- *1 Anwendungen sind in einer Messung nur Datenbank-Sätze zur Transaktionssteuerung
- *1 enthalten (Start, Finish, Cancel).
- *1 c) An der Programmschnittstelle der Administration werden für BeanConnect Proxy
- *1 Anwendungen bei Opcode KC_GET_OBJECT für TACs in der Struktur kc_tac_str auch
- *1 die Felder db_counter und db_elap_msec versorgt. Diese Information kann z.B. über
- *1 WinAdmin/WebAdmin ausgelesen werden.
- *1

2.2 Realisierte Change-Requests (CR)

Einige der im Kapitel 2.1 aufgeführten neuen Funktionen wurden aufgrund von Change-Requests realisiert:

- *1 • Lange Rechnernamen bis zu 64 Zeichen
- *1 • KDCEVAL – Ausgabe der TSN in der TRACE2-Liste

3 Technische Hinweise

3.1 Ressourcenbedarf

Plattenspeicherbedarf für die Installation:

<u>openUTM:</u>		
Solaris SPARC	32/64 Bit:	< 1 GB
Linux	32/64 Bit:	< 500 MB
HP-UX (Itanium)	32/64 Bit:	< 500 MB
AIX	32/64 Bit:	< 500 MB
Windows	32/64 Bit:	< 250 MB

Der Hauptspeicherbedarf für eine ablauffähige UTM-Anwendung ist zum einen abhängig von der Art der verwendeten Produkte, wie Sprachanschluss, Datenbank, X/Open-Schnittstelle und verteilte Verarbeitung, zum anderen von Menge und Umfang der Anwenderprogramme.

Der dynamische Speicherbedarf für UTM-Betriebsmittel zur Prozesskommunikation (Shared Memory) ist abhängig von Umfang und Parametrisierung der KDCDEF-Generierung.

3.2 SW-Konfigurationen

	Solaris SPARC	32/64 Bit	ab V10
	Linux(SuSE)	32/64 Bit	ab SLES 11 PL4
*1	Linux(RedHat)	32/64 Bit	ab RHEL 6.5
	HP-UX (Itanium)	32/64 Bit	ab V11.31
	IBM-AIX	32/64 Bit	ab V6.1
	Windows	32/64 Bit	ab Windows 8.x
	Windows Server	32/64 Bit	ab Windows Server 2012

UTM-Cluster-Anwendungen werden auf den Plattformen Solaris SPARC und Linux unterstützt. Auf Windows-Systemen sind UTM-Cluster-Anwendungen nicht allgemein freigegeben.

Die Knoten-Anwendungen einer UTM-Cluster-Anwendung müssen auf identischen Plattformen unter demselben Bit Mode ablaufen.

Für den Zugriff auf die globalen Cluster-Dateien von openUTM-Cluster-Anwendungen wird NFS V4 benötigt.

*1 Voraussetzung für die Nutzung der openSM2-Funktionalität ist der Einsatz von openSM2 ab V9.

Für die Kommunikation über TCP/IP wird mitgeliefert:

*1	PCMX(Solaris SPARC)	6.0B30
*1	PCMX(Linux i585)	6.0B30
*1	PCMX(Linux x86_64)	6.0B30
*1	PCMX(HP-UX Itanium)	6.0B30
*1	PCMX(AIX)	6.0B30
*1	PCMX-32(Windows)	5.0B20
*1	PCMX-64(Windows)	5.0B20

Der Einsatz von PCMX ist nur in Verbindung mit openUTM lizenzfrei.

Beim Einsatz von openUTM V6.5A müssen die auf der DVD mitgelieferten PCMX Versionen oder neuere eingesetzt werden.

*1 Für den Einsatz der openUTM Crypt Komponente auf Unix-/Linux- und Windows-Systemen ist die Software openssl V1.0 notwendig.

*1 Siehe auch www.openssl.org.

Für verteilte Transaktionsverarbeitung mit anderen UTM-Anwendungen werden folgende Versionen unterstützt:

openUTM(BS2000)	ab V6.3A
openUTM(Unix-, Linux-, Windows- Systeme)	ab V6.3A

Wenn es bei der Kopplung mit Versionen kleiner V6.3 Probleme gibt, dann besteht keine Wartungsverpflichtung seitens Fujitsu Technology Solutions; in diesem Fall sollte ein Kunde die Partneranwendungen auf UTM-Versionen größer/gleich V6.3 umstellen.

*1 Für die Kommunikation mit IBM-SNA-Systemen über LU6.2 können folgende Produkte eingesetzt werden:

openUTM-LU62(Unix- oder Windows-Systeme) V5.1

*1 und abhängig vom Betriebssystem die Fremdprodukte:

Solaris:

SNAP-IX ab Version 7.0.2.4; von Metaswitch Networks

Linux:

IBM Communications Server für Linux, ab Version 6.4

Windows:

IBM Communications Server für Windows, ab Version 6.4

Für verteilte Transaktionsverarbeitung mit Java EE Anwendungen werden folgende BeanConnect Versionen unterstützt:

BeanConnect ab V3.0 B

Für Client-Server-Kommunikation wahlweise:

openUTM-Client(Unix-, Linux- und Windows-Systeme) ab V6.3A
(enthält Trägersysteme UPIC V6.3; OpenCPIC V4.0)

openUTM-Client(BS2000) ab V6.3A

BeanConnect (Unix-, Linux- und Windows-Systeme) ab V3.0B
(enthält die Komponente openUTM-JConnect)

Für die Administration über grafische Oberflächen:

openUTM WinAdmin V6.5

openUTM WebAdmin V6.5

*1 Für alle Unix- und Linux- Systeme ist ein C/C++-Compiler erforderlich, für Windows wird Visual Studio
*1 2005/2008 oder höher benötigt. Der Einsatz des QuickStart Kit auf Windows setzt Visual Studio 2010
*1 voraus.

Auf Solaris-Systemen ist für den Einsatz der Beispiel-Anwendung Solaris Studio 12 erforderlich.

Hinweis für die Verwendung des Compilers XL C/C++ for AIX, V11.1:

Es gibt einen behindernden Fehler im Zusammenhang mit shared objects

<<http://www-01.ibm.com/support/docview.wss?uid=swg21443810>>

Für dieses Problem gibt es einen Patch von IBM, der mit "smit" eingespielt werden kann.

<<http://www-01.ibm.com/support/docview.wss?uid=swg24032451>>

COBOL-Compiler:

Zum Entwickeln von Cobol UTM-Anwendungen:

NetCOBOL ab V10.4 (Linux 64-Bit)

NetCOBOL ab V10.1 (Windows 32-Bit)

MicroFocus Visual COBOL ab V2.0

Zum Ablauf:

Cobol Runtime Lizenzen von Micro Focus

Hinweis: durch die Einführung der UTM-System-Prozesse kann es zu Problemen bei den Cobol-
Laufzeitlizenzen kommen, wenn diese zu knapp kalkuliert wurden.

Bei der Berechnung der benötigten Anzahl Lizenzen müssen die bis zu 3 zusätzlichen
Lizenzen mit berücksichtigt werden.

Auf Unix- und Linux-Systemen können folgende Datenbanksysteme eingesetzt werden:

ORACLE ab V11.2

INFORMIX Enterprise Edition ab V12.10

Auf Windows-Systemen können folgende Datenbanksysteme eingesetzt werden:

ORACLE ab V11.2

Hinweis:

Die Softwarekonfiguration enthält z.T. Produkte, die technisch einsetzbar sind, von denen es aber
bereits neuere Versionen gibt. Es sollten nach Möglichkeit immer die aktuellen Versionen eingesetzt
werden.

3.3 Produkt-Installation

3.3.1 Installation auf Unix-/Linux-Plattformen

Auf **Solaris-Systemen** mounten Sie die DVD. Im entsprechenden Dateiverzeichnis finden Sie die Produkte, die Sie mit „pkgadd“ installieren können:

```
pkgadd -d <DVD>/<package>.pkg
```

Auf **Linux-Systemen** wird das Standardverfahren „rpm“ verwendet:

```
rpm -i <DVD>/<package>.rpm --nodeps [--ignorearch] [--replacefiles]
      [--prefix=<install-dir>]
```

Zum Installieren von PCMX lesen Sie bitte die dazugehörige readme-Datei.

Zum Installieren von OUTM-LU62 lesen Sie bitte die dazugehörige readme-Datei.

Auf **HP-UX-Systemen** mounten Sie die DVD (mount -F cdcs -o cdcase <device> <mountpoint>).

Wechseln Sie in das Produkt-Dateiverzeichnis und starten Sie das Installationsskript mit:

```
./xpinstall.sh
```

Zum Installieren von PCMX verwenden Sie folgenden Aufruf:

```
swinstall -s <prod-dir>/cmx.ds -x target_type=tape -x reinstall=true CMX
```

Auf **AIX-Systemen** mounten Sie die DVD (mount -v cdrfs -r <device> <mountpoint>).

Wechseln Sie in das Produkt-Dateiverzeichnis und starten Sie das Installationsskript mit:

```
./xpinstall
```

Zum Installieren von PCMX verwenden Sie folgenden Aufruf:

```
installp -a -F -d <prod-dir>/cmx.ds CMX.all
```

Zum Installieren von OUTM-LU62 lesen Sie bitte die dazugehörige readme-Datei.

Auf allen Unix- und Linux-Systemen wird sowohl die 32-Bit als auch die 64-Bit Ablaufumgebung von openUTM installiert.

Der Einsatz von openUTM setzt die Kornshell (ksh) voraus.

Auf Linux-Systemen muss das Paket pdksh (Public Domain ksh) installiert sein.

3.3.2 Installation auf Windows-Plattformen

Auf Windows-Systemen gibt es separate Installationspakete von openUTM für die 32-Bit bzw. 64-Bit Ablaufumgebungen. Zum Installieren auf einer 32-Bit oder 64-Bit Ablaufumgebung wechseln Sie in das Produkt-Dateiverzeichnis und rufen unter Administrationsberechtigung „utm-32.msi“ oder „utm-64.msi“ bzw. „utm-crypt-32.msi“ oder „utm-crypt-64.msi“ auf.

*1

Die empfohlene Installationsreihenfolge ist:

1. Microsoft Visual C++
2. (optional) ausgewählter Cobol Compiler
3. PCMX (von openUTM-DVD)
4. openUTM

openUTM V6.5 darf ab Windows 8.x nicht in ein Verzeichnis, welches Leerzeichen enthält, wie z.B. „Program Files“, installiert werden, es muss ein anderes Verzeichnis verwendet werden.

Der Installationspfad darf eine Länge von 39 Zeichen nicht überschreiten.

3.3.3 Installation zusammen mit C++ Umgebung auf Unix-/Linux-Plattformen

Bei der Verwendung von C++-Teilprogrammen muss darauf geachtet werden, dass die C++ Entwicklungsumgebung von openUTM bei der Installation aktiviert werden kann.

Dazu gilt für den C++ Compiler folgendes:

- Solaris: type CC muss als Ergebnis den C++ Compiler liefern
- Linux: type g++ muss als Ergebnis den C++ Compiler liefern
- HP: der C++ Compiler muss unter /opt/aCC/bin/aCC zur Verfügung stehen
- AIX: der C++ Compiler muss unter /usr/vacpp/bin/cc zur Verfügung stehen

Bei der Installation von openUTM wird dann mainutmCC.o erstellt.

mainutmCC.o kann auch (falls erforderlich) nach der Installation von openUTM durch folgende Eingaben erstellt werden:

```
UTMPATH=<utmpfad>
<utmpfad>/shsc/CCmainutm
export UTMPATH
```

Näheres siehe Handbuch "Einsatz von UTM-Anwendungen".

*1 3.3.4 Installation zusammen mit openssl Umgebung auf Unix-/Linux-Plattformen

*1 Sie müssen openssl vor openUTM Crypt installieren. Bitte prüfen Sie nach der Installation von
*1 openssl, ob unter /usr/lib bzw. /usr/lib64 eine Bibliothek libcrypto.so existiert. Gibt es an deren Stelle
*1 nur eine Bibliothek namens libcrypto.so.<version>, so müssen Sie einen symbolischen Link aufsetzen:
*1 *ln -s libcrypto.so.<version> libcrypto.so*

*1 Bei der Installation von openUTM Crypt auf den Unix- und Linux-Plattformen wird die UTM Encryption
*1 Bibliothek mit der, auf der Zielplattform zur Verfügung stehenden, openssl Software gebunden. Dies
*1 erfolgt sowohl für die 32-Bit als auch die 64-Bit Ablaufumgebung.
*1 Kann bei einer der Bindeaktionen nicht auf die openssl Software zugegriffen werden, ist das Ergebnis
*1 der Installation „nicht erfolgreich“ und die Verschlüsselungsfunktionalität steht UTM nicht zur Verfügung.
*1 Bitte analysieren Sie den Inhalt der Bindeprotokolle libutmencrypt.so.out und libutmencrypt.so.err im
*1 Verzeichnis <utmpfad>/sys/crypt.

*1 Sobald die openssl Software zur Verfügung steht, können Sie die Binde-Aktion wiederholen:
*1 a) Melden Sie sich als root an und wechseln Sie in das UTM Installationsverzeichnis <utmpfad>.
*1 b) Führen Sie das Skript <utmpfad>/shsc/xpcrypt aus.
*1 c) Überprüfen Sie die Bindeprotokolle libutmencrypt.so.out und libutmencrypt.so.err im Verzeichnis
*1 <utmpfad>/sys/crypt.

3.3.5 Installationsprotokoll

Bei der Installation wird auf Unix-/Linux-Systemen eine Datei <utmpfad>/utm.log erzeugt. Hier wird die Information abgelegt, die während der Installation auf Terminal ausgegeben wird.

3.3.6 Hinweise: Kernelparameter für openUTM

Zum Betreiben einer UTM-Anwendung kann es notwendig sein, einige Unix-Kernelparameter zu vergrößern. Hiervon sind die folgenden Kernelparameter betroffen (ist mehr als ein Kernelparameter pro Zeile angegeben, so sind alle angegebenen Parameter auf denselben Wert zu ändern):

- MAXUP max. Anzahl Prozesse pro User
Generell kann es für einzelne Benutzer zu einem Prozessengpass kommen (Fehlermeldung: fork failed), wenn sie sehr viele Prozesse starten müssen. In diesem Fall ist MAXUP anzupassen.
- SEMMNS, SEMMNU max. Anzahl von Semaphoren und Semaphor-Strukturen
Die maximal benötigte Anzahl von Semaphoren lässt sich anhand der Beschreibung von

MAX SEMKEY im Handbuch "Anwendungen generieren" berechnen.

Es ist zu berücksichtigen, dass es eventuell weitere Produkte auf dem Rechner gibt, die ebenfalls Semaphore benötigen, z.B. werden für jede gestartete INFORMIX-Datenbank mindestens 20 Semaphore benötigt.

- SEMMAP, SEMMNI max. Anzahl von Semaphore-Identifizier
Die Semaphore werden zu Semaphore-Feldern zusammengefasst und jedem Semaphore-Feld ist ein (eindeutiger) Semaphore-Identifizier zugeordnet.
openUTM und Informix fassen jeweils 20 Semaphore zu einem Semaphore-Feld zusammen.
- SFSZLIM, HFSZLIM max. Größe einer Datei (in Byte)
Werte ab 48 MB (0x3000000) sind in der Regel auch für große UTM-DUMPs ausreichend.
In der Datei /etc/default/login ist der Eintrag ULIMIT=... ebenfalls an den neuen Wert anzupassen.
ULIMIT gibt die maximale Dateigröße (für normale User) in Blocks (1 Block = 512 Bytes) an.
- SDATLIM, HDATLIM max. Swap-Space pro Prozess (in Byte)
- SVMMLIM, HVMMLIM max. virtueller Speicher pro Prozess (in Byte)
Damit größere UTM-Anwendungen mit COBOL-Teilprogrammen noch gebunden werden können, müssen diese Werte auf mindestens 48 MB(0x3000000) erhöht werden, wobei darauf zu achten ist, dass der Swap-Bereich der Anlage >= 48 MB ist.

Änderung der Kernelparameter für Solaris-Systeme

Normalerweise ist es nicht erforderlich die Kernelparameter zu ändern. Es wird empfohlen, zum Tunen "/etc/project" zu verwenden. Details dazu entnehmen Sie bitte der Oracle Solaris Dokumentation.

Änderung der Kernelparameter für Linux-Systeme

Normalerweise ist es nicht erforderlich die Kernelparameter zu ändern. In welchen Dateien die Werte geändert werden können und wie Sie einen neuen Kernel erzeugen, entnehmen Sie bitte den Unterlagen Ihres Linux-Distributors.

Änderung der Kernelparameter für HP-UX-Systeme

Parameter	Define	Default	Empfehlung
max.Anzahl Prozesse/User	maxuprc	256	ausreichend
max.Anzahl Semaphore-Ids	semmni	64	ausreichend
max.Anzahl Semaphore	semmns	256	1024
max.Anzahl Semaphore-Strukturen	semmnu	30	200
	semume	10	66
Swap-Space	-	2*Mem	4*Mem

Es wird empfohlen zum Tunen das HP-Tool "/usr/bin/sam" zu verwenden. Siehe dazu das HP-Handbuch: System Administration Tasks - HP-9000 Series 800 Computers.

Änderung der Kernelparameter für AIX-Systeme

Parameter	Define	Default	Empfehlung
max.Anzahl Prozesse/User	maxuprc	256	ausreichend
max.Anzahl Semaphore-Ids	semmni	36	64
max.Anzahl Semaphore	semmns	128	256
Swap-Space	-	2*Mem	4*Mem

Es wird empfohlen zum Tunen das AIX-Tool "/usr/bin/smit" zu verwenden. Siehe dazu das AIX-Handbuch: System Management Guides - RS/6000.

3.4 Produkt-Einsatz

3.4.1 Hinweise zur Umstellung auf openUTM V6.5A

Bei Umstellung von der Vorgängerversion müssen

- KDCFILE und ROOT-Source erzeugt werden mit KDCDEF von V6.5A
- die ROOT-Source übersetzt werden
- das MSGTAC-Teilprogramm übersetzt werden, falls vorhanden
- die Source für das Meldungsmodul mit KDCMMOD erzeugt und dann übersetzt werden, falls ein eigener Meldungsmodul genutzt wird
- das Anwendungsprogramm neu gebunden werden
- die Anwenderdaten mit dem KDCUPD von openUTM V6.5A in die neue KDCFILE übertragen werden, wenn die Daten relevant sind

*1

KDCUPD unterstützt folgende Versionsübergänge:

openUTM(Unix, Linux, Windows-Systeme) V6.2A auf openUTM(Unix, Linux, Windows-Systeme) V6.5A
openUTM(Unix, Linux, Windows-Systeme) V6.3A auf openUTM(Unix, Linux, Windows-Systeme) V6.5A
openUTM(Unix, Linux, Windows-Systeme) V6.4A auf openUTM(Unix, Linux, Windows-Systeme) V6.5A
openUTM(Unix, Linux, Windows-Systeme) V6.5A auf openUTM(Unix, Linux, Windows-Systeme) V6.5A

Hinweis:

Eine Übertragung der Anwenderdaten mit KDCUPD vom 32-Bit-Modus in den 64-Bit-Modus ist immer nur ohne zusätzlichen Versionsübergang möglich, d.h. nur innerhalb der aktuellen Version.

3.4.2 Umstellung einer stand-alone Anwendung in eine UTM-Cluster-Anwendung

Bei einer UTM-Cluster-Anwendung sind folgende Umstellungen möglich:

- Umstellung einer stand-alone-Anwendung V6.5A auf eine UTM-Cluster-Anwendung V6.5A
- Umstellung einer UTM-Cluster-Anwendung V6.5A auf eine stand-alone UTM-Anwendung V6.5A

Eine detaillierte Beschreibung der Vorgehensweise finden Sie im Manual "Einsatz unter Unix-, Linux und Windows-Systemen".

3.4.3 Client-Programme mit lokaler Anbindung an openUTM

Client-Programme mit lokaler Anbindung an openUTM müssen neu übersetzt und gebunden werden.

3.4.4 Binden von UTM OSI-TP Anwendungen auf Unix- und Linux-Plattformen

Bei der Angabe der OSS Bibliotheken ist unbedingt die folgende Reihenfolge einzuhalten: Zuerst ist die Bibliothek libossutm.so und danach erst die Bibliothek liboss.so anzugeben.

Hält man die Reihenfolge nicht ein, dann führt das zu folgendem Start-Fehler:

P001 Fehler beim OSS Aufruf (o_create() ·): - 1, 300, 199, 0
K060 Der Anwendungslauf wurde abgebrochen; die Ursache ist XINI06.

3.4.5 Binden und Starten von UTM-Anwendungen auf Unix- und Linux-Systemen

Beim Binden und beim Starten einer UTM-Anwendung auf Unix- und Linux-Systemen muss ab openUTM V6.5 die Umgebungsvariable LD_LIBRARY_PATH bzw. LD_LIBRARY_PATH_64 auf <utmpfad>/sys gesetzt sein.

3.4.6 Binden von UTM-Anwendungen auf AIX-Systemen

Beim Binden einer UTM-Anwendung auf AIX-Systemen muss ab openUTM V6.5 zusätzlich die Bibliothek <utmpfad>/sys/libutmconvt.so angegeben werden.

*1 3.4.7 openUTM V6.5A KDCDEF ohne Encryption

*1 Beim Ablauf der UTM Generierung mit KDCDEF (ohne Encryption) wird in V6.5A00 die Meldung K508
*1 ausgegeben, wenn ein Partner vom Typ UPIC-R generiert ist; es sei denn, es wurde in der Generierung
*1 zusätzlich die Anweisung *OPTION GEN-RSA-KEYS=NO* angegeben.
*1 Bisher war dieses Verhalten zwar beschrieben, UTM hat aber auch im Falle von KDCDEF (ohne
*1 Encryption) die Meldung K508 nicht ausgegeben.

*1 3.4.8 Hinweise zum Einsatz von openUTM-CRYPT V6.5A

- *1 • Windows-Systeme
*1 Beim Ablauf des Dienstprogramms KDCDEF und beim Ablauf der UTM-Anwendung muss
*1 sichergestellt werden, dass die Bibliothek libeay32.dll gefunden wird. Ggfs. muss die PATH-
*1 Variable geeignet erweitert werden.
- *1 • AIX-Systeme
*1 Auf der Plattform AIX muss zusätzlich die LIBPATH Variable auf <utmpfad>/sys gesetzt werden.

3.4.9 Hinweise zur Beispielanwendung für UTM-Cluster-Anwendungen

Bei der Installation einer Beispielanwendung für eine UTM-Knoten-Anwendung muss folgendes berücksichtigt werden:

- Eine installierte UTM-Knoten-Anwendung kann nicht in eine stand-alone Anwendung konvertiert werden.
- Ein Filebase Verzeichnis für eine Knoten-Anwendung und ein Filebase Verzeichnis für eine stand-alone Anwendung dürfen im Dateibaum nicht unter demselben übergeordneten Dateiverzeichnis angelegt werden.

3.5 Entfallene und gekündigte Funktionen

3.5.1 Entfallene Funktionen

Mehrere Funktionen wurden gestrichen und können nicht mehr in KDCDEF generiert werden. Wenn sie dennoch angegeben werden, wird dies im KDCDEF-Lauf mit einem Syntaxfehler abgelehnt. Im Einzelnen wurde Folgendes gestrichen:

- Die Umwandlungsdatei zwischen kurzen und langen Rechnernamen (Mapping-Funktion) und die Umgebungsvariable UTM_NET_HOSTNAME werden nicht mehr unterstützt, da openUTM standardmäßig Rechnernamen bis zur Länge von 64 Zeichen unterstützt.
- KDCDEF-Anweisung MAX:
Operand NET-ACCESS= für die Einstellung der Netzanbindung (single-/multi-threaded). Es wird immer die multi-threaded Netzanbindung verwendet.
- KDCDEF-Anweisung FORMSYS für ein Formatierungssystem.
- KDCDEF-Anweisung OPTION:
Operand CHECKTNS= für den Abgleich der UTM-Generierung mit der TNS-Generierung.

3.5.2 Gekündigte Funktionen

- TNS Betrieb
In openUTM V6.5 wird die TNS Funktionalität letztmals unterstützt.
In Folgeversionen müssen bei openUTM die Adressierungsinformationen vollständig bei der Konfiguration mit KDCDEF hinterlegt werden.
- Encryption
Mit openUTM V6.5 werden die folgenden Encryption Level letztmalig unterstützt:
 - Encryption-Level 1 und 2 für PTERM und TPOOL
ENCRYPTION-LEVEL=1 (RSA-Schlüssellänge 200 Bit, DES-Verfahren)
ENCRYPTION-LEVEL=2 (RSA-Schlüssellänge 512 Bit, AES-Verfahren)
 - Encryption-Level 1 für TAC
Operanden-Wert *ENCRYPTION-LEVEL=1* für KDCDEF-Anweisung TAC
(Verschlüsselung nach dem DES-Verfahren)

3.6 Inkompatible Änderungen

- KDCS-Schnittstelle
 - Aufruf INFO SI
Die Länge der von dem KDCS-Aufrufs INFO SI zurückgegebenen Daten hat sich geändert. Bestehende Programme erhalten in UTM V6.5 den Return Code 01Z. Zur Vermeidung dieses Return Codes müssen diese Programme mit der Datenstruktur von UTM V6.5 neu übersetzt werden.
 - Aufrufe SIGN ON, CK und CP
Bei SIGN ON und SIGN CK muss der Parameter KCLA auf 16 (bisher 8) und bei SIGN CP auf 32 (bisher 16) gesetzt werden, ansonsten liefert der Aufruf den Return Code 43Z.
 - Aufruf DADM RQ
Bei DADM RQ muss der Parameter KCLA auf 54 (bisher 53) gesetzt werden, ansonsten liefert der Aufruf den Returncode 01Z.
- Administration
 - Programmschnittstelle allgemein:
Die Programmschnittstelle der Administration wurde inkompatibel geändert. Bestehende Programme sind davon jedoch nicht betroffen.
Werden Administrationsprogramme neu übersetzt und setzen *version_data* auf die aktuelle Version 10 (*KC_VERSION_DATA_10*), dann müssen bei Aufrufen zur Modifikation von Objekten jeweils die neuen Felder für die langen Rechnernamen versorgt werden. Ebenso müssen bei Aufrufen zum Lesen von Objekteigenschaften in der IdArea und der SelectionArea die Felder für die langen Rechnernamen versorgt werden. Betroffen sind die Objekte *KC_CON*, *KC_LSES*, *KC_PTERM*, *KC_TPOOL* und *KC_CLUSTER_NODE*.
Bei *KC_GET_OBJECT* wird für die Objekt-Typen *KC_LPAP*, *KC_OSI_CON*, *KC_PTERM* und *KC_TPOOL* in den Datenstrukturen *kc_lpap_str*, *kc_osi_con_str*, *kc_ptrm_str* und *kc_tpool_str* im Feld *map* der Wert ,1' statt ,S' für *SYS1/SYSTEM1/SYSTEM* zurückgegeben.
 - Programmschnittstelle *obj_type=KC_DB_INFO*:
Ab openUTM V6.5 müssen Sie bei Aufrufen an der Programmschnittstelle der Administration mit *opcode=KC_GET_OBJECT* und *obj_type=KC_DB_INFO id_lth =2* angeben. Bislang musste *id_lth = 0* angegeben werden.
 - Programmschnittstelle *obj_type=KC_USER*:
Administrationsprogramme, die für einen Benutzer ein neues, schon verschlüsseltes Passwort (*pw_encrypted = `Y / A`*) setzen, müssen die Version der Programmschnittstelle der Administration (*version_data*) auf die aktuelle Version 10

(KC_VERSION_DATA_10) setzen, und das neue verschlüsselte Passwort in das neue Feld *password16* schreiben.

- Meldungen
Die Meldungen K028 und K121 entfallen. Sie werden von openUTM V6.5 nicht mehr verwendet. Für die Meldungen K004 und K006 wurden Meldungstexte geändert.
- *1
*1
*1
*1
• Standardanmeldeverfahren und Zwischendialog eines Anmeldevorgangs
Das Standardanmeldeverfahren von UTM und der Zwischendialog eines Anmeldevorgangs wurden umgestellt, damit bei ungültigen Berechtigungsdaten möglichst wenig Information über die Gültigkeit von Benutzererkennung und Passwort nach außen gegeben wird.

Die Meldungen K004 und K006 können bei der Ausgabe auf die Ziele STATION, SYSOUT und SYSLST nicht mehr voneinander unterschieden werden, da die Texte dieser Meldungen angeglichen wurden. Auf den Meldungszielen MSGTAC, SYSLOG und USER-DEST können die Meldungen K004 und K006 auch weiterhin unterschieden werden.

Ist die Gültigkeit des Passworts abgelaufen und die Anwendung ist ohne Grace-Sign-On generiert, dann wird zuerst die Meldung K092 ausgegeben (bisher K120). Dies geschieht für Benutzer mit hellem oder mit dunklem Passwort, aber nur, wenn kein Anmeldevorgang aktiv ist. Nach der Passwort-Eingabe wird die Meldung K120 ausgegeben, falls das Passwort korrekt war. Andernfalls wird die Meldung K006 ausgegeben.

Ist die Gültigkeit des Passworts abgelaufen und die Anwendung ist mit Grace-Sign-On generiert, dann wird zuerst die Meldung K092 ausgegeben (bisher K155). Dies geschieht für Benutzer mit hellem oder mit dunklem Passwort. Für Benutzer mit dunklem Passwort geschieht dies auch im Zwischendialog eines Anmeldevorgangs. Wird das alte Passwort korrekt eingegeben, dann wird je nach Situation eine der Meldungen K097, K120, K125 oder K155 ausgegeben. Andernfalls wird die Meldung K006 ausgegeben.

Für Benutzer mit dunklem Passwort, für das die Gültigkeit in Kürze abläuft wird die Meldung K092 ausgegeben (bisher K121). Wird das alte Passwort korrekt eingegeben, dann wird die Meldung K122 zusammen mit der Meldung K008 ausgegeben. Der Benutzer hat dann erst beim nächsten Anmelden die erneute Gelegenheit das Passwort zu ändern.

Für Benutzer mit dunklem Passwort, für das die Mindestgültigkeit noch nicht erreicht ist, wird zur Passwort-Anforderung die Meldung K092 ausgegeben (bisher K028). Ändert ein Benutzer in der Meldung K092 das Passwort, obwohl dies zu diesem Zeitpunkt nicht zulässig ist, dann wird der Benutzer angemeldet und openUTM gibt zusammen mit der Meldung K008 die neue Meldung K112 (Das Passwort kann z.Zt. nicht geändert werden) aus.

Ist ein Benutzer gesperrt oder bereits angemeldet, dann wird immer erst das Passwort geprüft. Dazu wird ggfs. zuerst die Meldung K092 ausgegeben. Dies geschieht für Benutzer mit dunklem Passwort auch im Zwischendialog eines Anmeldevorgangs. Wird das Passwort korrekt eingegeben, dann wird je nach Situation eine der Meldungen K005 oder K007 ausgegeben. Andernfalls wird die Meldung K006 ausgegeben.

Ist die eingegebene Benutzererkennung in der Anwendung unbekannt, dann wird immer erst mit Meldung K092 zur Eingabe eines Passwort aufgefordert. Erst nach einer Passwort-Eingabe wird das Anmelden des unbekanntenen Benutzers abgelehnt.

Die Änderungen in den Meldungsausgaben beim Anmelden eines Benutzers müssen insbesondere von Programmen berücksichtigt werden, die eine Terminalschnittstelle emulieren.

- MSGTAC-Programm
Die Schnittstelle des MSGTAC-Programms wurde inkompatibel geändert. Felder, die Prozessornamen oder Passwörter enthalten, wurden auf 64 Bytes verlängert. Die MSGTAC-Anwenderprogramme müssen entsprechend angepasst werden.
Bei einer Auswertung der SYSLOG-Datei durch ein eigenes Programm müssen Anwender die geänderten MSGTAC-Strukturen berücksichtigen.

- Logging-Exit
Die Schnittstelle zum Logging-Exit wurde inkompatibel geändert. Die Felder, die Prozessornamen enthalten, wurden auf 64 Bytes verlängert.
Die Anwenderprogramme für den Logging-Exit müssen entsprechend angepasst werden.
- BCAM-Trace in openUTM
Die maximale Länge der Daten, die bei eingeschalteter BCAM-Tracefunktion aufgezeichnet werden können, wurde um 56 Bytes verkürzt. Der Maximalwert beträgt nun 32.624.
- Generierung
 - MAX TRMSGLTH
Unabhängig von dem generierten Wert für MAX TRMSGLTH nimmt KDCDEF für TRMSGLTH immer den Maximalwert von 32.700 an.
 - PRONAM und NETWORK-Selektor
Von KDCDEF wird ab openUTM V6.5 bei den Generierungsparametern
CON PRONAM=
PTERM PRONAM=
TPOOL PRONAM=
OSI-CON NETWORK-SELEKTOR=
die Angabe von Leerzeichen als Prozessornamen bzw. Network-Selektor nun auch, wie auf BS2000-Systemen, mit einer Fehlermeldung abgewiesen.
Zulässig ist die Angabe von Leerzeichen als Prozessornamen nur bei PTERM und TPOOL mit PTYPE = TTY, UPIC-L oder PRINTER.

3.7 Einschränkungen

3.7.1 Einschränkungen in UTM-Cluster-Anwendungen

keine

3.7.2 Allgemeine Einschränkungen

- Betreiben einer openUTM-Anwendung
Auf der Plattform Windows 64-Bit wird die PCMX Funktionalität TNS nicht mehr unterstützt. D.h. alle für die Kommunikation notwendigen Adressinformationen müssen bei der Generierung mit KDCDEF angegeben werden.
- Hinweise zum Einsatz der Beispielanwendung
Für die Beispielanwendung wird keine Gewährleistung und Service-Verpflichtung übernommen.
- Auf AIX-Systemen können - wie in den Vorgängerversionen zu openUTM V6.5A – die UTM-Systemmodule nur statisch eingebunden werden.

3.8 Verhalten im Fehlerfall

Zur Fehlerdiagnose sind folgende Angaben nötig:

- genaue Beschreibung der Fehlersituation
- Angabe der Versions-/Korrekturstände der beteiligten Software
- Genaue Angabe des Rechnertyps

Die Fehlerunterlagen sollten möglichst vollständig vorhanden sein. Als Fehlerunterlagen werden benötigt:

- UTM-Dumps aller Work-Prozesse sowie zugehörige gcores. Diese Dateien sollten als Binärdateien, d.h. in nicht aufbereiteter Form, vorliegen
- SYSLOG-Datei(en)
- stdout/stderr-Protokolle der UTM-Prozesse
- KDCDEF-Generierung, Startprozedur und Startparameter
- core-Dateien mit zugehörigen Phasen (utmwork) und shared objects. Die shared objects können ermittelt werden mit "ldd utmwork"
- Binder- und Übersetzungslisten

Bei Fehlern in einer UTM-Cluster-Anwendung zusätzlich:

- alle Cluster globale Dateien, Protokolle (und DUMPS) aller Knoten-Anwendungen
- bei Problemen, die durch das Zusammenspiel der Knoten-Anwendungen verursacht wurden, die Protokolldateien von allen anderen Knoten-Anwendungen
- die Startprozedur und die bei der Generierung als EMERGENCY-CMD und FAILURE-CMD angegebenen Prozeduren

Es soll versucht werden, die Fehler unter Verwendung statischer Bibliotheken zu reproduzieren.

Vorgehen bei Fehlern:

- PENDER-Dump auswerten
- Fehler reproduzieren unter Verwendung geeigneter Debugger wie z.B. dbx, gdb
- Aufrufhierarchie beim core mit Hilfe eines Debuggers ermitteln

Bei abnormalem Anwendungsabbruch müssen die UTM-Dumps und gcores aller Prozesse gesichert werden. Falls möglich sollte der Abbruch mit TESTMODE=ON (Startparameter) reproduziert werden.

Abbruch mit Signalen:

Trat ein PENDER-Dump mit 70Z/XT10 oder XT11 oder Anwendungsabbruch mit SIG010/SIG011 (Signal SIGBUS/SIGSEGV) auf, sollte die UTM-Signalbehandlung mit dem Startparameter START STXIT=OFF ausgeschaltet werden, damit beim fehlerhaften Befehl sofort ein core-Dump erzeugt wird. Vor dem nächsten Neustart der Anwendung muss auf jeden Fall kdcrem aufgerufen werden, um evtl. noch vorhandene Semaphore und Shared Memories sowie Statusinformationen der Anwendung in der <utmpfad>/applifile zu löschen bzw. zurückzusetzen.

Bei Fehlern, die in Zusammenhang mit der openUTM-Netzanbindung stehen, können zusätzlich folgende Unterlagen erstellt werden:

- Meldungen der UTM-Netzprozesse in stdout/stderr
- CMX-Traces
- OSS-Traces
- UTM-BCAM-Trace
- Umwandlungsdatei mit Mapped Hostname Einträgen
- Bei heterogener Kopplung Generierungsinformation und Trace von openUTM LU6.2 und seiner SNA Komponenten und vom IBM-SNA System

Erstellung der UTM-BCAM-Traces und OSS-Traces siehe Handbuch "Meldungen, Test und Diagnose".

Bei Fehlern, die in Zusammenhang mit Datenbanken stehen, können zusätzlich folgende Unterlagen erstellt werden:

- XA Debug Trace
- weitere Unterlagen siehe entsprechende Freigabemitteilung des Datenbanksystems

Nach Startfehlern (K049-Meldung) z.B. mit Insert 40 / 32 muss vor einem neuen Start das UTM-Tool kdcrem aufgerufen werden.

4 Hardware-Unterstützung

Es wird die Hardware unterstützt, auf der die genannten Betriebssystemversionen ablauffähig sind. Eingeschlossen sind alle auf der Intel-x86-Technologie basierenden Systeme z.B. Laptops, PCs, PRIMERGY-Systeme; SPARC-Systeme, z.B. SPARC Enterprise Server und weitere Unix-Systeme: IBM pSeries; ia64 HP Server, z.B. HP rx2800; weitere Plattformen auf Anfrage.

CPU mindestens 250 MHz, Hauptspeicher mind. 512 MB

UTM-Cluster-Anwendungen:

- Für den Betrieb von UTM-Cluster-Anwendungen ist ein Network File System/Service (NFS) notwendig
- openUTM (Solaris-Systeme):
Knoten einer UTM-Cluster-Anwendung können Systeme mit unterschiedlichen Solaris-Versionen, aber einheitlicher Adressierung (entweder 64-Bit oder 32-Bit) sein. Knoten mit anderen Betriebssystemen (Linux, BS2000) sind nicht möglich
- openUTM (Linux-Systeme):
Knoten einer UTM-Cluster-Anwendung können Systeme mit unterschiedlichen Linux-Versionen, aber einheitlicher Adressierung (entweder 64-Bit oder 32-Bit) sein. Knoten mit anderen Betriebssystemen (Unix-Systeme, andere Linux-Distributionen, BS2000) sind nicht möglich