IBM Tivoli Netcool/OMNIbus SNMP Probe Helm Chart 2.0.0

Reference Guide
February 28, 2019

SC27-9507-01

Note

Before using this information and the product it supports, read the information in Appendix A, “Notices and Trademarks,” on page 13.

Edition notice

This edition (SC27-9507-00) applies to version 2.0.0 of IBM Tivoli Netcool/OMNIbus SNMP Probe Helm Chart and to all subsequent releases and modifications until otherwise indicated in new editions.

This edition replaces SC27-9507-01.

Contents

About this guide

The following sections contain important information about using this guide.

Document control page

Use this information to track changes between versions of this guide.

The SNMP Probe Helm Chart documentation is provided in softcopy format only. To obtain the most recent version, visit the IBM® Tivoli® Knowledge Center:

https://www.ibm.com/support/knowledgecenter/SSSHTQ/omnibus/helms/common/Helms.html

Table 1. Document modification history

Document version	Publication date	Comments
SC27-9507-00	August 9, 2018	First IBM publication.
SC27-9507-01	February 28, 2019	Guide updated for version 2.0.0 of the helm chart. Helm chart now supports ICP 3.1.x. The following topics were updated: • "Obtaining the PPA package" on page 1 • "Prerequisites" on page 1 • "Resources required" on page 2 • "Uninstalling the chart" on page 4 • "Configurable parameters" on page 4 • "Limitations" on page 11 The following topics were added: • "PodSecurityPolicy requirements" on page 2 • "Troubleshooting" on page 12

Chapter 1. SNMP Probe Helm Chart

The SNMP Probe Helm Chart allows you to deploy a cluster of SNMP Probes onto Kubernetes. These probes process SNMP notifications or traps from managed devices or SNMP agents to a Netcool Operations Insight operational dashboard.

This guide contains the following sections:

The Knowledge Center contains the following additional topics that contain information that is common to all Helm Charts:

- Specifying the image repository
- Loading PPA packages to IBM Cloud Private
- Exposing the probe service
- Upgrading to a new version of the probe helm charts
- Changing the service type during a helm upgrade

Obtaining the PPA package

You can download the installation package from the **IBM Passport Advantage** website.

Use the **Find by part number** field to search for the following part number: `CC0F9EN`

Chart details

The chart deploys the Tivoli Netcool/OMNIbus SNMP Probe onto Kubernetes to receive SNMP notifications or traps. The probe deployment is fronted by a service.

This chart can be deployed more than once on the same namespace.

Prerequisites

This solution requires the following applications:

- IBM Tivoli Netcool/OMNIbus ObjectServer to be created and running prior to installing the probe. To create and run the IBM Tivoli Netcool/OMNIbus ObjectServer, see the following topic on the IBM Knowledge Center: Creating and running ObjectServers.
- Netcool Knowledge Library (NcKL) Intra-Device correlation automation is installed and enabled. This automation creates the following objects in the ObjectServer to aid in determining the causal releavence of events:
  - Intra-device correlation (AdvCorr) tables within the alerts database.

  - Supplementary automations implemented as an AdvCorr trigger group and three related triggers.
  - Additional columns in the `alerts.status` table.
- Kubernetes 1.11.1.
- Tiller 2.9.1

Note : Operator role is a minimum requirement to install this chart.

The chart must be installed by a Administrator to perform the following tasks:

- Enable Pod Disruption Budget policy when installing the chart.
- Retrieve sensitive information from a secret such as SNMP v3 Users data.

The chart must be installed by a Cluster Administrator to perform the following tasks in addition to those listed above:

- Obtain the Node IP using kubectl get nodes command if using the NodePort service type.
- Create a new namespace with custom PodSecurityPolicy if necessary. For details see "PodSecurityPolicy requirements" on page 2.

Resources required

This solution requires the following resources:

- CPU Requested : 250m (250 millicpu)
- Memory Requested : 256Mi (~ 268 MB)

PodSecurityPolicy requirements

This chart requires a PodSecurityPolicy to be bound to the target namespace prior to installation. You can choose either a predefined PodSecurityPolicy or have your cluster administrator create a custom PodSecurityPolicy for you.

The predefined PodSecurityPolicy name `ibm-restricted-psp` has been verified for this chart, see IBM Cloud Pak Pod Security Policy Definitions. If your target namespace is bound to this PodSecurityPolicy, you can proceed to install the chart. The predefined PodSecurityPolicy definitions can be viewed here: https://github.com/IBM/cloud-pak/blob/master/spec/security/psp/README.md

This chart also defines a custom PodSecurityPolicy which can be used to finely control the permissions/capabilities needed to deploy this chart. You can enable this custom PodSecurityPolicy using the ICP user interface or the supplied instructions/scripts in the `pak_extension pre-install` directory. For detailed steps on creating the PodSecurityPolicy see https://www.ibm.com/support/knowledgecenter/SSHTQ/omnibus/helms/all_helms/wip/reference/hlm_common_psp.html

From the user interface, you can copy and paste the following snippets to enable the custom Pod Security Policy

- From the user interface, you can copy and paste the following snippets to enable the custom PodSecurityPolicy:
  - Custom PodSecurityPolicy definition:

```
    apiVersion: extensions/v1beta1
    kind: PodSecurityPolicy
    metadata:
      annotations:
        kubernetes.io/description: "This policy is based on the most restrictive policy,
        requiring pods to run with a non-root UID, and preventing pods from accessing the
host."
        seccomp.security.alpha.kubernetes.io/allowedProfileNames: docker/default
        seccomp.security.alpha.kubernetes.io/defaultProfileName: docker/default
      name: ibm-netcool-probe-snmp-prod-psp
    spec:
      allowPrivilegeEscalation: false
      forbiddenSysctls:
```

```
    - '*'
    fsGroup:
      ranges:
      - max: 65535
        min: 1
      rule: MustRunAs
    hostNetwork: false
    hostPID: false
    hostIPC: false
    requiredDropCapabilities:
    - ALL
    runAsUser:
      rule: MustRunAsNonRoot
    seLinux:
      rule: RunAsAny
    supplementalGroups:
      ranges:
      - max: 65535
        min: 1
      rule: MustRunAs
    volumes:
    - configMap
    - emptyDir
    - projected
    - secret
    - downwardAPI
    - persistentVolumeClaim
```

– Custom ClusterRole for the custom PodSecurityPolicy:

```
apiVersion: rbac.authorization.k8s.io/v1
kind: ClusterRole
metadata:
  name: ibm-netcool-probe-snmp-prod-clusterrole
rules:
- apiGroups:
  - extensions
  resourceNames:
  - ibm-netcool-probe-snmp-prod-psp
  resources:
  - podsecuritypolicies
  verbs:
  - use
```

– RoleBinding for all service accounts in the current namespace. Replace {{ NAMESPACE }} in the template with the actual namespace:

```
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: ibm-netcool-probe-snmp-prod-rolebinding
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: ClusterRole
  name: ibm-netcool-probe-snmp-prod-clusterrole
subjects:
- apiGroup: rbac.authorization.k8s.io
  kind: Group
  name: system:serviceaccounts:{{ NAMESPACE }}
```

- From the command line, you can run the setup scripts included under `pak_extensions`.

  As a cluster administrator, the pre-install scripts and instructions are in the following location:

  `pre-install/clusterAdministration/createSecurityClusterPrereqs.sh`

  As team admin/operator the namespace scoped scripts and instructions are in the following location:

  `pre-install/namespaceAdministration/createSecurityNamespacePrereqs.sh`

Installing the chart

To install the chart, use the following steps:

1. Extract the helm chart archive and customize `values.yaml`. The configuration section lists the parameters that can be configured during installation.
2. Install the chart with the release name `my-snmp-probe` using the configuration specified in the customized `values.yaml` using following command:

   ```
   helm install --namespace <your pre-created namespace> --name my-snmp-probe -f values.yaml stable/ibm-netcool-probe-snmp-prod --tls
   ```

   Where: *my-snmp-probe* is the release name for the chart.

   Helm searches for the `ibm-netcool-probe-snmp-prod` chart in the helm repository called `stable`. This assumes that the chart exists in the stable repository.

Tip : You can list all releases using `helm list --tls` or you can search for a chart using `helm search`.

The command deploys the probe on the Kubernetes cluster using a default configuration. For a list of the parameters that you can configure during installation see "Configurable parameters" on page 4.

Verifying the chart

See the instructions at the end of the helm installation for chart verification. The instructions can also be displayed by viewing the installed helm release under **Menu -> Workloads -> Helm Releases** or by running the following command:

```
helm status <release> --tls
```

Uninstalling the chart

To uninstall or delete the chart, use the following command:

```
helm delete my-snmp-probe --purge --tls
```

Where: *my-snmp-probe* is the release name for the chart.

The command removes all the Kubernetes components associated with the chart and deletes the release.

Clean up any prerequisites that were created

As a Cluster Administrator, run the cluster administration cleanup script included under `pak_extensions` to clean up cluster scoped resources when appropriate.

```
post-delete/clusterAdministration/deleteSecurityClusterPrereqs.sh
```

As a Cluster Administrator, run the namespace administration cleanup script included under `pak_extensions` to clean up namespace scoped resources when appropriate.

```
post-delete/namespaceAdministration/deleteSecurityNamespacePrereqs.sh
```

Configuring the chart

The integration requires configuration of the chart parameters.

Configurable parameters

You use parameters to specify how the probe interacts with the device. You can override the chart's default parameter settings during installation.

The following table describes the configurable parameters for this chart and lists their default values.

Table 2. Configurable parameters	
Parameter name	**Description**
license	The license state of the image being deployed. Enter `accept` to install and use the image. The default value is `not accepted`
replicaCount	The number of deployment replicas generated. This parameter is omitted when **autoscaling.enabled** is set to `true` The default value is `1`
global.image.secretName	The name of the secret containing the docker config to pull the image from a private repository. Leave this parameter blank if the probe image already exists in the local image repository or the Service Account has a been assigned with an Image Pull Secret. There is no default value set for this parameter.
image.repository	Use this parameter to specify the probe image repository. Update this repository name to pull from a private image repository. The image name should be set to `netcool-probe-snmp` The default value is `netcool-probe-snmp`
image.tag	Use this parameter to specify the `netcool-probe-snmp` image tag. The default value is `20.2.0_4`
image.testRepository	Use this parameter to specify the utility image (busybox) repository. Update this repository name to pull from a private image repository. The default value is `busybox`
image.testImagetag	Use this parameter to specify the utility image tag. The default value is `1.28.4`
image.pullPolicy	The image pull policy. The default value is `IfNotPresent`
netcool.primaryServer	The primary Netcool/OMNIbus server to connect to. This is usually set to NCOMS or AGG_P. The default value is `nil`
netcool.primaryHost	Use this parameter to specify the host of the primary Netcool/OMNIbus server. Specify the ObjectServer hostname or IP address. The default value is `nil`

Table 2. Configurable parameters (continued)	
Parameter name	**Description**
netcool.primaryPort	The port of the primary Netcool/OMNIbus server. The default value is `nil`
netcool.backupServer	The backup Netcool/OMNIbus server to connect to. If the **backupServer**, **backupHost** and **backupPort** parameters are defined in addition to the primaryServer, primaryHost, and primaryPort parameters, the probe will be configured to connect to a virtual ObjectServer pair called `AGG_V`. The default value is `nil`
netcool.backupHost	The host of the backup Netcool/OMNIbus server. Specify the ObjectServer hostname or IP address. The default value is `nil`
netcool.backupPort	The port of the backup Netcool/OMNIbus server. The default value is `nil`
probe.messageLevel	The probe log message level. The default value is `warn`
probe.rulesFile	The type of probe rules file to use. Set this parameter to `NCKL` to use the Netcool Knowledge Library Rules Files pre-installed in the `netcool-probe-snmp` image. The default value is `Standard`
probe.snmpv3. snmpConfigChangeDetectionInterval	The frequency (in minutes) that the probe checks for `mttrapd.conf` configuration changes. Specify a value between 0 and 10080. Setting this value to `0` disables automatic detection and loading. The default value is `1`
probe.snmpv3.snmpv3Only	Set this value to `true` to process only SNMPv3 traps and informs. This allows you to limit event processing. If you set this value to `false` the probe processes SNMP v1, v2, and v3. The default value is `false`
probe.snmpv3.reuseEngineBoots	Specifies whether the probe reuses the engine ID and the number of SNMP engine boots specified in the `mttrapd.conf file`. Set this value to `true` to reuse the engine ID and number of SNMP boots. If you set this value to `false`, the probe does not reuse the engine ID. The default value is `true`

Table 2. Configurable parameters (continued)

Parameter name	Description
probe.snmpv3.usmUserBase	Specifies whether the probe reads the `mttrapd.conf` file in the directory specified by the **PeristentDir** property, the **ConfPath** property, or both of these directories. This parameter takes the following values: `0`: The probe uses only the file in the **ConfPath** directory. 1: The probe uses only the file in the **PersistenDir** directory. 2: The probe uses both files. The default value is 2
probe.snmpv3.snmpv3MinSecurityLevel	Specifies which SNMPv3 traps the SNMP Probe processes. This parameter takes the following values: 1: The probe processes SNMP V3 traps and inform PDUs of security level `NoAuth`, `AuthNoPriv`, or `AuthPriv`. 2: The probe processes SNMP V3 traps and inform PDUs of security level `AuthNoPriv` or `AuthPriv`. 3: The probe processes SNMP V3 traps and inform PDUs of security level `AuthPriv`. The default value is `1`
probe.snmpv3.secretName	Name of the existing secret containing an encoded list of USM user configuration. Leave this parameter unset to create a new secret with the user settings configured in **probe.snmpv3.users**. The default value is `nil`
probe.snmpv3.users	List of security users for SNMP v3. This value is ignored if **probe.snmpv3.secretName** is set. For details about how to configure this property using JSON see .“SNMP V3 security user configuration” on page 9. The default value is `[]`
service.probe.type	The SNMP Probe k8 service type exposing ports. The default value is `ClusterIP`
service.probe.externalPort	The external TCP and UDP port for this service. The default value is 162

Table 2. Configurable parameters (continued)

Parameter name	Description
autoscaling.enabled	Set this parameter to `false` to disable auto-scaling. The default value is `true`
autoscaling.minReplicas	The minimum number of probe replicas. The default value is 2
autoscaling.maxReplicas	The maximum number of probe replicas. The default value is 5
autoscaling.cpuUtil	The target percentage CPU utilization. For example, enter `60` for `60%` target utilization. The default value is `60`
poddisruptionbudget.enabled	Set this parameter to `true` to enable Pod Disruption Budget to maintain high availability during node maintenance. Administrator role or higher is required to enable Pod Disruption Budget on clusters with Role Based Access Control. The default value is `false`. The default value is `false`
poddisruptionbudget.minAvailable	The minimum number of available pods during node drain. This can be set to a number or a percentage, for example: 1 or 10%. **CAUTION :** Setting this parameter to 100%, or to the number of replicas, may block node drains entirely. The default value is `1`
resources.limits.memory	The container memory limit. The default value is `512Mi`
resources.limits.cpu	The container CPU limit. The default value is `500m`
resources.requests.cpu	The container CPU requested. The default value is `250m`
resources.requests.memory	The container memory requested. The default value is `256Mi`
arch	The worker node architecture. This is fixed to amd64.

SNMP V3 security user configuration

An administrator can create a secret prior to installing the chart. The chart can then be configured to use this existing secret by specifying the secret name in probe.snmpv3.secretName parameter.

For details about creating a secret, see "Creating a secret with SNMP v3 users data" on page 10.

To create a new secret automatically during chart installation, leave the **probe.snmpv3.secretName** parameter unset and follow the details below to set the **probe.snmpv3.users** parameter to specify a list of SNMP V3 users.

The SNMP V3 User object consists of the following parameters.

Table 3. Configurable parameters

Parameter name	Description
name	The security user name. Example: `netcoolTrap`
authEncryptionMethod	The authentication type (`MD5`, `SHA`, or `SHA256`). When running in FIPS 140-2 mode, use SHA for this parameter. Example: `MD5`
authEncryptionPassword	The authentication password. This must be at least eight characters in length. Example: `tr4psMD5`
privacyEncryptionMethod	The type of privacy (either `DES` or `AES`). When running the probe in FIPS 140-2 mode, use AES for this parameter. This parameter is optional. Example: `DES`
privacyEncryptionPassword	The privacy password. This parameter is optional. Example: `tr4psDES`
authEngineIdentifier	The engine ID of the trap source associated with the user. The engine ID is required for traps, but optional for informs. Example: `0x0102030405`

The example settings above are shown below. They should be set in the **probe.snmpv3.users** parameter and the **probe.snmpv3.secretName** should be unset to enable them.

```
- name: netcoolTrap
              authEncryptionMethod: MD5
              authEncryptionPassword: tr4psMD5
              privacyEncryptionMethod: DES
              privacyEncryptionPassword: tr4psDES
              authEngineIdentifier: '0x0102030405'
```

You can add more user entries to add more security users. The following example shows two security users `netcoolTrap` and `netcoolInforms`.

Note : Use the `values.yaml` file to specify the list of users especially when you intend to add multiple security users.

```
probe:
  snmpv3:
    enabled: true
    users:
      - name: netcoolTrap
        authEncryptionMethod: "MD5"
        authEncryptionPassword: "tr4psMD5"
        privacyEncryptionMethod: "DES"
        privacyEncryptionPassword: "tr4psDES"
        authEngineIdentifier: "0x0102030405"
      - name: netcoolInform
        authEncryptionMethod: "MD5"
        authEncryptionPassword: "1nformsMD5"
        privacyEncryptionMethod: "DES"
        privacyEncryptionPassword: "1nformsDES"
```

Creating a secret with SNMP v3 users data

This section shows how to create a secret with the sample SNMP V3 user settings (specified in JSON) below. This setting contains two users netcoolTrap and netcoolInform, which is the same as the example in SNMP V3 Security User Configuration.

```
- name: netcoolTrap
  authEncryptionMethod: MD5
  authEncryptionPassword: tr4psMD5
  privacyEncryptionMethod: DES
  privacyEncryptionPassword: tr4psDES
  authEngineIdentifier: '0x0102030405'
- name: netcoolInform
  authEncryptionMethod: MD5
  authEncryptionPassword: 1nformsMD5
  privacyEncryptionMethod: DES
  privacyEncryptionPassword: 1nformsDES
  authEngineIdentifier: ''
```

1. For the user setting above, the entries that needs to be used are shown below. Save these entries in a file.

```

```

2. Save the above entries into a file called users.txt.

```
$ cat <<EOF >> users.txt
> createUser -e 0x0102030405 netcoolTrap MD5 tr4psMD5 DES tr4psDES
> createUser netcoolInform MD5 1nformsMD5 DES 1nformsDES
> EOF

$ cat users.txt
createUser -e 0x0102030405 netcoolTrap MD5 tr4psMD5 DES tr4psDES
createUser netcoolInform MD5 1nformsMD5 DES 1nformsDES
```

3. Encode the contents of users.txt using Base64 encoding. Note: When using the base64 utility on Darwin/macOS users should avoid using the -b option to split long lines. Conversely Linux users should add the option -w 0 to base64 commands or the pipeline base64 | tr -d '\n' if -w option is not available.

```
$ base64 users.txt
Y3JlYXRlVXNlciAtZSAweDAxMDIwMzA0MDUgbmV0Y29vbFRyYXAgTUQ1IHRyNHBzTUQ1IERFUyB0cjRwc0RFUwpjcmVhd
GVVc2Vy
IG5ldGNvb2xJbmZvcm0gTUQ1IDFuZm9ybXNNRDUgREVTIDFuZm9ybXNERVMK
```

4. Then insert the base64 encoded string into a secret.yaml file with mttrapd.conf as the key as shown below. This file will create a new Kubernetes secret with the called my-snmp-probe-snmpv3-users

```
# Secrets created separately from the release
apiVersion: v1
kind: Secret
```

```
metadata:
  name: my-snmp-probe-snmpv3-users
type: Opaque
data:
  mttrapd.conf :
Y3JlYXRlVXNlciAtZSAweDAxMDIwMzA0MDUgbmV0Y29vbFRyYXAgTUQ1IHRyNHBzTUQ1IERFUyB0cjRwc0R
FUwpjcmVhdGVVc2VyIG5ldGNvb2xJbmZvcm0gTUQ1IDFuZm9ybXNNRDUgREVTIDFuZm9ybXNERVMK
```

5. Use the following command to create the secret on Kubernetes in the default namespace and to verify that the secret is created correctly.

```
$ kubectl -n default apply -f secret.yaml
secret "my-snmp-probe-snmpv3-users" created

$ kubectl get secrets my-snmp-probe-snmpv3-users --namespace default -o yaml
apiVersion: v1
data:
  mttrapd.conf:
Y3JlYXRlVXNlciAtZSAweDAxMDIwMzA0MDUgbmV0Y29vbFRyYXAgTUQ1IHRyNHBzTUQ1IERFUyB0cjRwc0RF
UwpjcmVhdGVVc2VyIG5ldGNvb2xJbmZvcm0gTUQ1IDFuZm9ybXNNRDUgREVTIDFuZm9ybXNERVMK
kind: Secret
metadata:
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: |
      {"apiVersion":"v1","data":
{"mttrapd.conf":"Y3JlYXRlVXNlciAtZSAweDAxMDIwMzA0MDUgbmV0Y29vbFRyYXA
gTUQ1IHRyNHBzTUQ1IERFUyB0cjRwc0RFUwpjcmVhdGVVc2VyIG5ldGNvb2xJbmZvcm0gTUQ1IDFuZm9ybXNNRDUgREVT
IDFuZm9
ybXNERVMK"},"kind":"Secret","metadata":{"annotations":{},"name":"my-snmp-probe-snmpv3-users",
"namespace":"default"},"type":"Opaque"}
  creationTimestamp: 2018-07-20T03:54:14Z
  name: my-snmp-probe-snmpv3-users
  namespace: default
  resourceVersion: "2520884"
  selfLink: /api/v1/namespaces/default/secrets/my-snmp-probe-snmpv3-users
  uid: 91a034ec-8bd0-11e8-983d-005056a0a011
type: Opaque
```

6. With the secret above created, set the SNMP Probe **`probe.snmpv3.secretName`** to `my-snmp-probe-snmpv3-users` to use the pre-created secret.

Limitations

This solution has the following limitations:

- Only the AMD64 / x86_64 architecture is supported for IBM Tivoli Netcool/OMNIbus SNMP Probe.
- It is validated to run on IBM Cloud Private 3.1.0 and 3.1.1.
- The NcKL rules files are pre-built in the `netcool-probe-snmp` image and are not customizable.
- Due to a limitation on Kubernetes Ingress resource, additional post-installation step is required in order to receive external TCP/UDP traffic when using ClusterIP service type. A Cluster Administrator needs to reconfigure the nginx-ingress-controller with a "static" configuration based on Configmaps and restart the ingress controller for the changes to take effect. CAUTION: Restarting the ingress controller would impact other workloads running. Consider performing the change during a planned downtime in production environments. For more details see Exposing the probe service.

For details about the SNMP Probe, see https://www.ibm.com/support/knowledgecenter/en/SSSHTQ/omnibus/probes/snmp/wip/concept/snmp_introduction_c.html. For details about the NcKL, see https://www.ibm.com/support/knowledgecenter/en/SSSHTQ/omnibus/probes/nckl/wip/reference/nckl_intrdctn.html.

Troubleshooting

The following table describes how to troubleshoot issues when deploying the chart and how to resolve them.

Table 4. Problems

Problem	Cause	Resolution
The probe logs show an error when loading or reading rules files. Failed during field verification check. Fields `CorrScore`, `AdvCorrCauseType`, `CauseType`, `LocalObjRelate`, and `RemoteObjRelate` not found.	The NcKL intra-device correlation automation is not installed, and so the required fields are missing.	Install the NcKL intra-device correlation automation in your ObjectServer and redeploy the chart.

Appendix A. Notices and Trademarks

This appendix contains the following sections:

- Notices
- Trademarks

Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing 2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
Software Interoperability Coordinator, Department 49XA

3605 Highway 52 N
Rochester, MN 55901
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs. © Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Trademarks

IBM, the IBM logo, ibm.com, AIX, Tivoli, zSeries, and Netcool are trademarks of International Business Machines Corporation in the United States, other countries, or both.

Adobe, Acrobat, Portable Document Format (PDF), PostScript, and all Adobe-based trademarks are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

Intel, Intel Inside (logos), MMX, and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

Linux is a trademark of Linus Torvalds in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

SC27-9507-01