

# 使用dumpcfg管理工具跟踪ICM配置更改

## Contents

[Introduction](#)

[Prerequisites](#)

[Requirements](#)

[Components Used](#)

[Conventions](#)

[背景信息](#)

[转存Config Message Log](#)

[Related Information](#)

## [Introduction](#)

本文描述dumpcfg实用工具程序，是数据库管理工具。dumpcfg工具的主要功能将转存记录器数据库的Config\_Message\_Log表和显示表在一种有意义的格式。Config\_Message\_Log表记录在IP联络中心(IPCC)企业环境的配置更改。

## [Prerequisites](#)

### [Requirements](#)

Cisco 建议您了解以下主题：

- Cisco智能联络管理(ICM)
- Cisco IPCC

### [Components Used](#)

本文档中的信息基于以下软件和硬件版本：

- Cisco IPCC企业版

The information in this document was created from the devices in a specific lab environment.All of the devices used in this document started with a cleared (default) configuration.If your network is live, make sure that you understand the potential impact of any command.

### [Conventions](#)

Refer to [Cisco Technical Tips Conventions](#) for more information on document conventions.

## [背景信息](#)

ICM系统记录插入、删除和更新在配置数据库。系统在名为Config\_Message\_Log的数据库表里保留此配置历史记录。

Config\_Message\_Log是存储配置消息的数据库系统表。Config\_Message\_Log包含这些元素：

1. **RecoveryKey** : RecoveryKey是ICM软件内部使用跟踪虚拟时间的值。此键可以被转换成十二位配置序号，由一个增加每个新建记录的。
2. **LogOperation** : LogOperation指示配置更改的种类。“请添加”，并且“更新”是LogOperation的示例。
3. **TableName** : TableName表示配置更改影响表的名字。
4. **DateTime** : 当一套消息被记录了时，DateTime指示日期和时间。
5. **ConfigMessage** : ConfigMessage列出在处理的所有配置消息。

## 转存Config\_Message\_Log

Config\_Message\_Log表是相当隐秘在本地状态。所以，为了了解Config\_Message\_Log表，您需要析取数据，并且转换数据成一种可读和有意义的格式。dumpcfg工具能处理数据和显示在此列表的信息：

- 谁执行了什么操作
- 当完成了更改请发生
- 哪个程序做了变动

在Config\_Message\_Log表里为了查看详细资料，您能运行dumpcfg实用工具程序。配置更改审计线索出现。您能执行dumpcfg工具记录器数据库。这是此工具语法：

```
dumpcfg <database>[@server] <low recovery key> [high recovery key]
```

在此语法：

- <database >表示记录器数据库的区分大小写的名称，例如， lab60\_sideA。
- **@server**表示主机名- ICM管理工作站(AW)或日志记录器。**Note:** 如果运行at命令正在考虑中的服务器，请替代与‘@’字符的@server。
- **<低恢复键>**和[high recovery key]表示配置序号(RecoveryKey字段在Config\_Message\_Log表里)。为了跟踪所有更改， ICM增加此配置序号(请参阅桃红色长方形在[表1](#))。

运行此命令生产这些更改一完全转储：

```
dumpcfg lab60_sideA@. 0
```

调整lowRecoveryKey的值为零(0)，并且下降highRecoveryKey (请参阅箭头A在[表1](#))。

[图1](#)显示dumpcfg程序的部分输出。

为了运行dumpcfg和存储输出在ASCII文件命名了dumpcfg-output.txt，运行命令以此语法：

```
dumpcfg lab60_sideA@. 0 > dumpcfg-output.txt
```

图1 – dumpcfg工具和部分输出

```
C:\>dumpcfg lab60_sideA0. 0 ← A
[ 411867067000.0, First, Apr 18 2006 5:01PM ]
msgTransactionID: 12188

[ 411867067001.0, Add, Apr 18 2006 5:01PM ]
PID: 3708
MachineName: HPIPCCHDS
UserDomain: ICM60
UserName: khamb
ProcessName: conicrList(upcc)
SQLServerUserName: dbo

[ 411867067002.0, Add, User_Group, Apr 18 2006 5:01PM ]
ObjectAccessLevel: 0
UserGroupID: 5069
CustomerDefinitionID: 0
UserGroupType: U
ServiceProvider: Y
ReadOnly: N
FeatureSetID: 0
ChangeStamp: 0
UserGroupName: ICM60_test
Description: test

[ 411867067003.0, Add, User_Group_Member, Apr 18 2006 5:01PM ]
ObjectAccessLevel: 0
UserGroupMemberID: 5069
ChangeStamp: 0
UserGroupName: ICM60_test
UserName: ICM60_test

[ 411867067004.0, Last, Apr 18 2006 5:01PM ]
msgTransactionID: 12188

[ 411867067005.0, First, Apr 21 2006 3:39PM ]
msgTransactionID: 12190

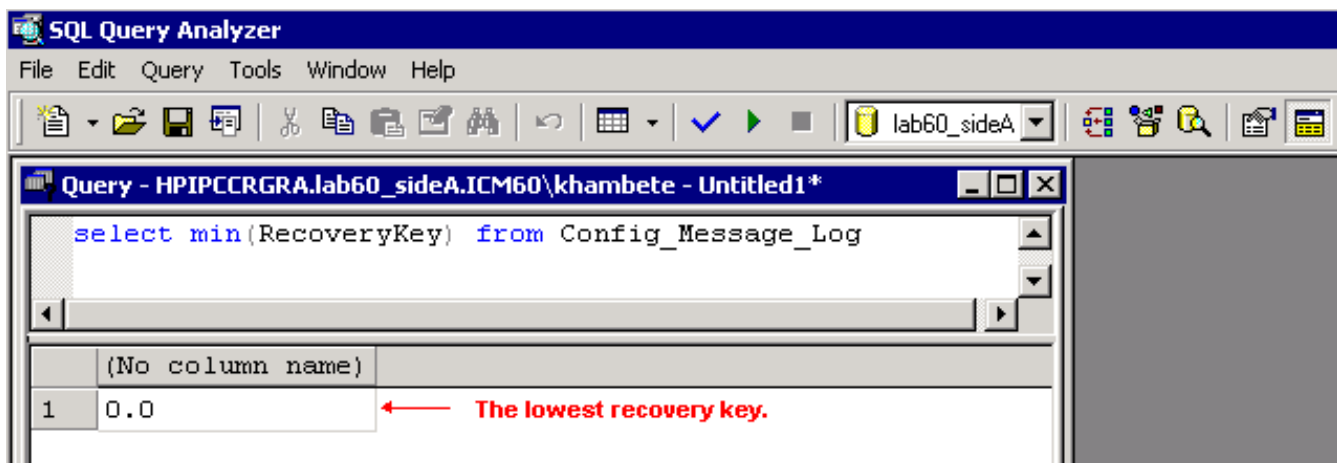
[ 411867067006.0, Add, Apr 21 2006 3:39PM ]
PID: 3572
MachineName: HPIPCCHDS
UserDomain: ICM60
UserName: bphi
ProcessName: scripted(upcc)
SQLServerUserName: dbo
```

您能确定从配置的实际序号在这些方式之一中：

- 运行这些SQL命令：为了确定最低的恢复键：

```
select min(RecoveryKey) from Config_Message_Log
```

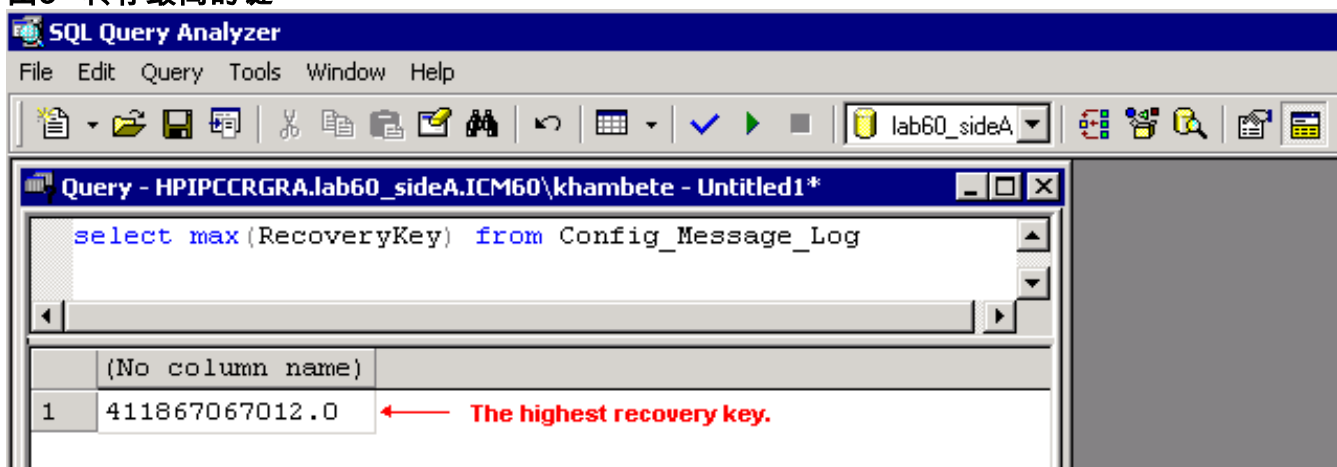
图2 –转存最低的恢复键



为了确定最高的恢复键：

```
select max(RecoveryKey) from Config_Message_Log
```

图3 – 转存最高的键



为了指定DateTime获得在某一日期前后发生或计时的更改：

```
select RecoveryKey from Config_Message_Log where
  DateTime >= '4/15/2006 00:00'
```

- 查看注册路由器或日志记录器。每当您插入，更新或删除配置或写脚本，路由器或日志记录器在日志包括新的序号：

```
select RecoveryKey from Config_Message_Log where
  DateTime >= '4/15/2006 00:00'
```

## [Related Information](#)

- [Technical Support & Documentation - Cisco Systems](#)