

Использование служебной программы командной строки OPCSTest

Содержание

[Введение](#)

[Предварительные условия](#)

[Требования](#)

[Используемые компоненты](#)

[Условные обозначения](#)

[Запуск opctest](#)

[команды opctest](#)

[Данные отладки](#)

[Выход и завершение сеанса opctest](#)

[Дополнительные сведения](#)

[Введение](#)

Служебная программа командной строки Open Peripheral Controller (OPC) Test (opctest) позволяет просматривать и настраивать различные параметры в OPC-процессе периферийного шлюза (PG) интеллектуального управления контактами (ICM) Cisco. Программу opctest можно запустить в командной строке ОС Microsoft Windows или из сеанса Telnet.

Примечание: Используйте командную строку, или в ПК или через rsANYWHERE. Использование командной строки предпочтительнее, поскольку отображение столбцов данных часто занимает больше места, чем может показать Telnet.

[Предварительные условия](#)

[Требования](#)

Для этого документа отсутствуют особые требования.

[Используемые компоненты](#)

Настоящий документ не имеет жесткой привязки к каким-либо конкретным версиям программного обеспечения и оборудования.

[Условные обозначения](#)

[Дополнительные сведения об условных обозначениях см. в документе Условные](#)

Запуск opctest

Выполните следующие действия, чтобы запустить opctest:

1. Выполните команду `opctest /cust custname /node ICM node name`. **Примечание:** `custname` соответствует имени пользователя, а `ICM node name` — имени узла. Пример: `opctest /cust abc /node pgx`.
2. Выполните команду `opctest /?` из командной строки. Эта команда отображает сведения о выполнении `opctest`.
3. После запуска `opctest` введите `help` или `?`, чтобы просмотреть список всех доступных команд. Наиболее часто используемая команда – `status`, которая отображает работоспособность и состояние PG.

Пример выходных данных:

```
C:\> opctest /? Version: Release 4.0, Build 04624 Usage: opctest [/f InputFile] [/system SystemName] [/cust Customer] [/node ICRNode] [/pipe OutputPipe] [/debug] [/stop] [/help] [/?]
```

[На рис. 1](#) отображены более подробные выходные данные команды `status`:

Рис. 1 — Выходные данные команды `status` OPCTest

The screenshot shows the output of the `opctest /cust/node pgl a` command. The output includes version information, current and local times, and the status of various processes and agents. Annotations with arrows point to specific parts of the output:

- An arrow points to the `OPC Sync:` line, with a note: "Indicates which side of the PG is supplying the Call Router with status of the PG".
- An arrow points to the process list, with a note: "Processes on the PGs & their states.".
- An arrow points to the `B pgag` process, with a note: "PGAG – manages session layer communications between the PG & the Central Controller SEE NOTE BELOW".
- An arrow points to the `PGAgent` table, with a note: "Peripheral Interface Manager/PIM".
- An arrow points to the `CTIServerNo` table, with a note: "CTI Server".

```
C:\>opctest /cust/node pgl a
OPCTEST Release 2.5 (service pack 2), Build 03105
opctest: status
OPC Version: Release 2.5 (service pack 2), Build 03116
Release Date: 09/28/98 07:01:57

Current Time: 03/17 17:47:07
Local Time: 03/17 12:47:07 (5.0 hr)
OPC Up: 12/08 18:59:52 (98.9 day)

OPC Sync: 03/13 22:18:33 (3.8 day) (A->B)

Process LastStateChange LastHeartBeat
A opc H-- 03/17 17:46:52 (16 sec)
A pgag OK M- 12/08 18:59:56 (98.9 day) --
A piml OK M- 03/12 19:35:58 (4.9 day) --
A ctisvr --- --
B pgag OK M- 12/08 18:59:53 (98.9 day) --
B piml OK M- 03/13 22:18:42 (3.8 day) --
B ctisvr --- --
B opc H-- 03/17 17:46:52 (16 sec)

PGAgent LastStateChangeTime ConnectATime Status ConnectBTime Status
SideA PIA 03/13 22:18:32 (3.8 day) 03/13 22:18:32 (3.8 day) CONNECTED 03/13 22:18:32 (3.8 day) CONNECTED
SideB P-- 03/13 22:18:32 (3.8 day) 03/13 22:18:32 (3.8 day) CONNECTED 03/13 22:18:32 (3.8 day) CONNECTED

PeripheralID Side State LastStateChange LastHeardFrom
1 A PIM_ACTIVE PR 03/13 22:18:32 (3.8 day) 03/17 17:47:07 (1 sec)

CTIServerNo Side State LastStateChange LastHeardFrom
1 ? CTI_NULL 12/08 19:00:02 (98.9 day) --
```

Примечание: В версии ICM 4.1 Peripheral Gateway Agent (PGAgent) раздел только

отображает для текущей активной стороны. В этом примере активной стороной является PGAgent на PG5B. PG5A бездействует:

```
PGAgent      LastStateChangeTime      ConnectATime      Status      ConnectBTime
Status
SideA  P-- 02/01 11:50:23 (3.2 hr)      IDLE AGENT
IDLE AGENT
SideB  PIA 02/01 11:48:54 (3.2 hr)      02/01 11:48:54 (3.2 hr) CONNECTED 02/01 11:48:54 (3.2
hr) CONNECTED
```

команды opctest

Введите `help` или `?` в командной строке `opctest`, чтобы просмотреть список доступных команд.

Некоторые из команд `opctest`, например `List_Agents` и `List_Trunk_Group`, требуют один или несколько дополнительных ключей командной строки. Введите `command name /?`, чтобы получить правильный синтаксис. Например:

```
opctest: la
list_agents: Error for PeripheralID: Missing argument.

opctest: la /?
Usage: list_agents PeripheralID [/agent AgentID] [/state AgentState]
      [/group SkillGroupID] [/agpri SkillGroupPriority] [/logout]
      [/help] [/?]

opctest: la 5004
SkillGroup=    0 Pri= 0 ----- LoggedOn=23 Avail=0 NotReady=4 Ready=19 TalkingIn=16
                TalkingOut=0 TalkingOther=3 WorkRdy=0 WorkNRdy=0 Busy=0
Reserved=0 Hold=0
SkillGroup=    1 Pri= 0 ----- LoggedOn=9 Avail=0 NotReady=1 Ready=8 TalkingIn=7
                TalkingOut=0 TalkingOther=1 WorkRdy=0 WorkNRdy=0 Busy=0
Reserved=0 Hold=0
SkillGroup=    2 Pri= 0 ----- LoggedOn=25 Avail=0 NotReady=4 Ready=21 TalkingIn=20

opctest: ltg
list_trunk_groups: Error for PeripheralID: Missing argument.

opctest: ltg 5004
Perph#   SkTargetID NTGSSkTargetID NumTrunks  LastHHU      Tracing  Ext
ConfigParam
   0      5057      5005             -1 02/01 14:30:00  0
   1      5058      5005             -1 02/01 14:30:00  0
   2      5059      5005             -1 02/01 14:30:00  0
   3      5060      5005             -1 02/01 14:30:00  0
   4      5061      5005             -1 02/01 14:30:00  0
   5      5062      5005             -1 02/01 14:30:00  0
   6      5063      5005             -1 02/01 14:30:00  0
   7      5064      5005             -1 02/01 14:30:00  0
   8      5065      5005             -1 02/01 14:30:00  0
   9      5066      5005             -1 02/01 14:30:00  0
  10      5067      5005             -1 02/01 14:30:00  0
  12      5010      5005             -1 02/01 14:30:00  0
  13      5011      5005             -1 02/01 14:30:00  0
  14      5068      5005             -1 02/01 14:30:00  0
```

Данные отладки

Выполните команду `debug`, чтобы включить специальную отладку в пределах `opctest`.

Команда `debug` включает управление отладкой с помощью усиленной трассировки.

`EMSTraceMask OPC-`. Включение управления отладкой активирует трассировку для части OPC, требующей дополнительной трассировки. Результат трассировки отображается в файлах журналов системы управления событиями (EMS) OPC. Используйте `dumplog` для просмотра выходных данных журналов EMS. [Дополнительные сведения см. в Использование служебной программы Dumplog.](#)

Например:

```
opctest: debug /? Usage: debug_control [/realtime] [/agent] [/halfhour] [/rcmeter] [/routing]
[/skillgroup] [/closedcalls] [/cstaecr] [/cstacer] [/pimmsg] [/ctimsg] [/rcmsg] [/dmpmsg]
[/icmsg] [/opcmsg] [/mdsmg] [/pdmsg] [/inrcmsg] [/passthru] [/tpmsg] [/physctrlr] [/periph]
[/all] [/help] [/?]
```

Выполните команду `debug /routing`, если требуется устранить неполадку, связанную с маршрутом трансляции.

После завершения устранения неполадки выполните команду с ключом `/noall`, чтобы отключить все трассировки OPC. Если оставить трассировку включённой, может заметно снизиться производительность.

[Выход и завершение сеанса opctest](#)

Выполните команду `quit`, чтобы выйти из служебной программы `opctest`.

Внимание. : Проявите осмотрительность при запуске команды `exit_opc`. Эта команда даёт инструкцию OPC-процессу выполнить выход на обеих сторонах PG в случае дуплексной передачи. Диспетчер Node Manager принудительно перезапускает процесс, который затем принудительно перезагружает конфигурацию для маршрутизатора вызова. Состояния всех внутренних периферийных устройств и агента сбрасываются. Затем OPC и диспетчер периферийных интерфейсов (PIM) вновь собирают данные о PG и его конфигурации.

[Дополнительные сведения](#)

- [Методы использования служебной программы создания дампа журнала](#)
- [Усиление трассировки](#)
- [Cisco Systems – техническая поддержка и документация](#)