

# Как Запустить Сервер с Отладки jvm, но от Демона Монитора.

## Содержание

[Введение](#)

[Вопрос. . Как запустить сервер с отладки jvm, но от демона монитора.](#)

## Введение

Этот документ описывает специальные рабочие режимы сервисов CIS.

## Вопрос. . Как запустить сервер с отладки jvm, но от демона монитора.

В некоторых ограниченных случаях и только в рекомендации Поддержки DV Cisco, вы, возможно, должны выполнить сервер в jvm режиме отладки для длительного периода времени. Этот определенный режим отладки является тем, где сервер запущен с этой команды:

`#!/composite_server.sh` отладка

Это запускает Сервер информации о Cisco как приоритетный процесс в режиме отладки (порт прослушивания отладки по умолчанию = 8000). Процесс Демона Монитора Cisco не запущен в этом случае. Однажды в этом режиме, поддержка Cisco захочет подключить отладчик к вашему порту 8000.

**Примечание:** Этот режим отладки не является тем же как более обычным типом Отслеживания отладки, где вас просят включить различные уровни отладки от `/conf/server/log4j.properties` файла или включить отладку от UI конфигурации Studio. Только выполните это, когда проинструктировано Поддержкой Cisco.

### Проблема для рассмотрения в этом случае:

Если вы случайно закрываете сеанс или отменяете процесс с `cntl-c`, при начале сервера как это процесс сервера работает на переднем плане и мог быть завершен. Если необходимо оставить сервер, работающий в этом режиме для длительных периодов времени, это может быть опасно для отъезда сеанса открытым. Кроме того, без демона монитора, работающего, это не будет присутствовать для перезапуска сервера автоматически, если это остановится по какой-либо причине.

Инструкции, чтобы избежать оперативного режима и обычно запускать сервер:

1) в `/conf/server/server.properties` отредактируйте эту линию (зависит от os, который вы используете. здесь, Linux используется в качестве примера и **полужирным**),

```
linux.vargs = "-сервер-XX:NewRatio=6-XX:-UseGCOverheadLimit - XX: +  
HeapDumpOnOutOfMemoryError-xx:heapdumpath =/opt/Cisco_Systems/CIS_7.0.3_brad/logs-  
xx:permsize=64m-xx:maxpermsize=256m-xx:-reduceinitialcardmarks - XX: +  
ExplicitGCInvokesConcurrent - XX: + UseConcMarkSweepGC"
```

2) Вставьте эту линию (показанный полужирным)

```
>-Xdebug-Xnoagent-Djava.compiler=NONE-Xrunjdp:transport=dt_socket, server=y, suspend=n,  
address=8000
```

в vmargs.

В данном примере это вставлено незадолго до предельных двойных кавычек.

```
linux.vargs = "-сервер-XX:NewRatio=6-XX:-UseGCOverheadLimit - XX: +  
HeapDumpOnOutOfMemoryError-xx:heapdumpath =/opt/Cisco_Systems/CIS_7.0.3_brad/logs-  
xx:permsize=64m-xx:maxpermsize=256m-xx:-reduceinitialcardmarks - XX: +  
ExplicitGCInvokesConcurrent - XX: + UseConcMarkSweepGC-Xdebug-Xnoagent-  
Djava.compiler=NONE-Xrunjdp:transport=dt_socket, server=y, suspend=n, address=8000
```

Это теперь позволяет серверу работать в режиме отладки и заставлять его также слушать на порту 8000 для наших средств отладки, которые вы будете использовать позже.

Результат сервера совпадет с при выполнении **отладки/bin/composite\_server.sh**, кроме теперь, с этой server.properties модификацией, можно обычно запускать сервисы путем призыва нашего демона монитора, вместо того, чтобы выполнить **composite\_server.sh** файл.

3) Запускайте сервер обычно, как этот **bin/composite.sh**, который запускает монитор

4) Проверьте, что сервер работает (может войти с Studio для тестирования),

5) Позвольте Поддержке Cisco использовать средство отладки для присоединения к серверу.