

mCore.MKD

Мультисервисный коммутатор доступа

**Подсистема Alarm Processor: Переменные и
SNMP-трапы**

РУКОВОДСТВО ПОЛЬЗОВАТЕЛЯ

Авторские права

Без предварительного письменного разрешения, полученного от НТЦ «ПРОТЕЙ», этот документ и любые выдержки из него, с изменениями и переводом на другие языки, не могут быть воспроизведены или использованы.

Оглавление

1 Общие сведения.....	4
1.1 Техническая поддержка.....	5
2 Назначение и основные свойства mCore.MKD.....	6
2.1 Основные сведения о подсистеме Alarm Processor.....	6
2.2 Использование протокола SNMP в подсистеме Alarm Processor.....	7
3 Настройка подсистемы Alarm Processor.....	8
3.1 Файл конфигурации ar.cfg.....	9
3.2 Алгоритм формирования идентификатора трапа.....	11
4 Использование SNMP-менеджеров.....	13
5 Описание переменных и трапов mCore.MKD.....	15
5.1 Переменные.....	15
5.2 Трапы.....	22
6 Приложение.....	30
Пример реального файла конфигурации ar.cfg.....	30

1 Общие сведения

Документ содержит описание переменных и трапов подсистемы Alarm Processor mCore.MKD. Также приведены рекомендации действий обслуживающего персонала при получении Alarm-сообщений, информирующие об изменении состояния оборудования или нарушения его работоспособности.

1.1 Техническая поддержка

Техническая поддержка, а также дополнительное консультирование по вопросам, возникающим в процессе установки и эксплуатации изделия, осуществляются производителем и службой технической поддержки.

Производитель

НТЦ «ПРОТЕЙ»

194044, Санкт-Петербург

Большой Сампсониевский пр., д. 60, лит. А

Бизнес-центр «Телеком СПб»

Тел.: (812) 449-47-27

Факс: (812) 449-47-29

WEB: <http://www.protei.ru>

E-mail: info@protei.ru

Служба технической поддержки

НТЦ «ПРОТЕЙ»

194044, Санкт-Петербург

Большой Сампсониевский пр., д. 60, лит. А

Бизнес-центр «Телеком СПб»

Тел.: (812) 449-47-27 доп. 5999 (круглосуточно)

(812) 449-47-31

Факс: (812) 449-47-29

WEB: <http://www.protei.ru>,

<http://www.imak.ru>

E-mail: mak.support@protei.ru

2 Назначение и основные свойства mCore.MKD

mCore.MKD - это программный коммутатор, относящийся к оборудованию операторского класса и используемый в качестве управляющего узла городских, сельских или производственных телефонных сетей.

Основная задача программных коммутаторов (Softswitch) — это управление вызовами — поиск и предоставление вызывающей стороне информации о точках соединения, используя которую оборудование вызывающего и вызываемого абонентов будет способно установить соединение.

mCore.MKD поддерживает базовые услуги и широкий набор дополнительных услуг (переадресация, постановка на ожидание и т.д.), включая контроль доступа абонентов к местной/ междугородной/международной телефонной связи.

2.1 Основные сведения о подсистеме Alarm Processor

Подсистема Alarm Processor — это подсистема мониторинга состояния аппаратных и логических ресурсов mCore.MKD.

В подсистеме Alarm Processor реализовано два способа предоставления информации:

- по запросу оператора (синхронный способ);
- посылка Alarm-сообщения при возникновении события (асинхронный способ).

Подсистема Alarm Processor взаимодействует с SNMP-менеджером, установленным на компьютере оператора, с использованием протокола SNMP. SNMP-менеджер предоставляет оператору удобные средства взаимодействия с подсистемой Alarm Processor и наглядные способы отображения состояния контролируемого оборудования. В качестве SNMP-менеджера может быть использована любая существующая программа, поддерживающая протокол SNMP.

Для SNMP-протокола любое оборудование представляет собой набор переменных, через которые SNMP-менеджер получает информацию о состоянии оборудования, и изменяет характер поведения оборудования через изменения значения этих переменных.

В mCore.MKD набор SNMP-переменных является отражением компонентной архитектуры программного обеспечения. Все программные компоненты в mCore.MKD имеют иерархическую зависимость между собой, образуя в совокупности дерево компонентов. Каждый компонент имеет уникальное имя (адрес), представляющее собой путь от корня дерева к компоненту, состоящее из массива слов, разделенных точками.

Каждый компонент в подсистеме Alarm Processor может зарегистрировать свой набор переменных. Переменные, зарегистрированные компонентом, делятся на два типа: обычная переменная и траповая переменная.

При изменении значения обычной переменной со стороны mCore.MKD, никакие события не активизируются. Просмотр обычных переменных происходит по запросу оператора, т.е. синхронным способом.

Траповая переменная способна активизировать событие (трап) при изменении своего значения. Траповая переменная посылается по SNMP-протоколу на компьютер оператора асинхронно, т.е. по факту появления данного события. Трап имеет свойство — приоритет, отображающий важность события. Другой свойство трапа — идентификатор. Идентификатор трапа — это уникальное число в пределах mCore.MKD. По идентификатору трапа SNMP-менеджер выполняет поиск трапа в mib-файле, если поиск завершился успешно, считывается информация о трапе, далее SNMP-менеджер выполняет действия согласно данной информации (например, определяет каким цветом вывести сообщение о трапе на экран, или какие действия предпринять на компьютере оператора: подача звукового сигнала, вывод окна с информационным сообщением и т.д.).

Для того, чтобы различать обычные и траповые переменные, в составе свойств переменных имеется свойство - «индикатор трапа». Индикатор трапа — это флаг, который установлен в единицу, если переменная является траповой.

Адреса переменных образуются из адреса компонента, которому принадлежит переменная и имени самой переменной. Например, адрес переменной, содержащей компонентный адрес MCU «MKD.Mod_MCU.MCU;CA».

Адрес переменной для передачи по протоколу SNMP имеет несколько иную форму — это набор чисел, разделенных точками, например, «1.2.3.100.1». Поэтому перед отправкой переменной SNMP-менеджеру, ее адрес преобразуется из компонентного адреса в SNMP-адрес. Правила преобразования из компонентного адреса в SNMP-адрес находятся в файле конфигурации /usr/protei/MKD/MKD/config/Alarm/ap.cfg, содержащий кроме этого и другие параметры настройки подсистемы Alarm Processor.

2.2 Использование протокола SNMP в подсистеме Alarm Processor

Простой протокол сетевого управления (SNMP) – это протокол управления компонентами сети. Протокол SNMP позволяет выполнять мониторинг текущего состояния отдельных компонентов сети, а также позволяет выполнять изменение параметров компонентов сети, изменяя таким образом характер поведения данных сетевых компонентов.

Основная идея протокола SNMP — это то, что мониторинг состояния сетевого компонента и управление им, производятся через набор переменных, хранимых в самом устройстве, - в Административной Базе Данных (MIB). Например, для того, чтобы проконтролировать состояние сетевого компонента, необходимо получить доступ к его MIB, и проанализировать значения интересующих переменных. Таким образом снимается зависимость протокола SNMP от конкретной реализации оборудования, делая его универсальным средством

На текущий момент протокол SNMP является фактически стандартом при реализации процесса мониторинга состояния сетевого оборудования и управления его параметрами. Использование протокола SNMP подсистемой Alarm Processor в mCore.MKD обеспечило совместимость со множеством существующих программ, предназначенных для управления сетевыми компонентами (SNMP-менеджеры). Одной из наиболее известных таких программ является программа SNMPc Castle Rock.

В mCore.MKD для протокола SNMP не поддерживается возможность управления через изменение значений переменных, реализован только мониторинг состояния логических и аппаратных ресурсов.

Использование протокола SNMP для мониторинга состояния mCore.MKD, дает

следующие возможности:

1. Получение в режиме реального времени состояния аппаратных и логических ресурсов.
2. Посылка уведомительных сообщений (trap) при изменении состояния аппаратных или логических ресурсов.
3. Настройка условий формирования уведомительных сообщений.
4. Использование программ SNMP-менеджеров сторонних разработчиков.

Полный SNMP-адрес переменной можно разделить на две части: идентификатор предприятия изготовителя сетевого компонента и идентификатор переменной в пределах сетевого компонента. Идентификатор предприятия-изготовителя является отражением глобальной иерархической структуры и будет неизменным для всех продуктов производства НТЦ «Протей». Идентификатор переменной в пределах сетевого компонента является отражением иерархической структуры аппаратных и логических ресурсов конкретного компонента. Таким образом достигается уникальность SNMP-адреса переменной. Т.е., любая SNMP-переменная является частью дерева SNMP-переменных в пределах сетевого компонента, это дерево является частью глобального дерева.

Идентификацией предприятия в составе глобального дерева занимается специальная международная организация, определяющая SNMP-адрес корневого узла дерева переменных для данного предприятия. Для «НТЦ Протей» определен SNMP-адрес корневого узла дерева переменных — «1.3.6.1.4.1.20873».

3 Настройка подсистемы Alarm Processor

Для настройки подсистемы Alarm Processor существуют два файла:

- ar.cfg — файл конфигурации подсистемы Alarm Processor ;
- ar_dictionary — словарь подсистемы Alarm Processor.

Файлы ar.cfg и ar_dictionary находятся в разделе /usr/protei/MKD/MKD/config/Alarm/.

Файл ar.cfg содержит параметры подсистемы Alarm Processor, параметры SNMP-соединения и правила преобразования компонентных адресов в SNMP-адреса.

В файле ar_dictionary находятся соответствия между значениями переменных и идентификаторами трапов. Идентификаторы трапов используются SNMP-менеджером для соответствующей их обработки.

Пример файла конфигурации ar.cfg приведен в «Приложении».

3.1 Файл конфигурации ar.cfg

В таблице ниже приведено описание секций, из которых состоит файл ar.cfg.

Таблица. Состав секций файла ar.cfg.

Имя секции	Описание
General	Основные параметры.
AtePath2ObjName	Правила преобразования компонентного адреса переменной в SNMP-адрес.
SNMP	Сетевые параметры протокола SNMP.
StandartMib	Объекты стандартного MIB-а.
SNMPTrap	Правила посылки трапов.
Filter	Правила фильтрации Alarm-сообщений.
SpecificTrapCA_Object	Соответствие идентификатора трапа адресу компонента.
SpecificTrapCT_Object	Соответствие идентификатора трапа типу компонента (CT).
SpecificTrapCA_Var	Соответствие идентификатора трапа компонентному адресу переменной.
Logs	Параметры ведения журналов подсистемой Alarm Processor.

В таблице ниже описаны параметры секции **[General]**.

Параметры секции [General] не требует какого-либо редактирования.

Таблица. Параметры секции [General].

Имя параметра	Описание
ApplicationAddress	Адрес приложения (по умолчанию «MKD.201», изменять не рекомендуется).
MaxConnectionCount	Максимальное кол-во одновременных подключений к AP_Agent (по умолчанию — 100).
ManagerThread	Флаг: 0 — подсистема Alarm Processor запускается в главном потоке приложения; 1 — подсистема Alarm Processor запускается в отдельном потоке. По умолчанию — 0.
CyclicWalkTree	Флаг: 0 — не выполнять циклический обход дерева компонентов; 1 — выполнять циклический обход дерева компонентов. По умолчанию — 1. Параметр изменять не рекомендуется.

Значение параметра ManagerThread зависит от загрузки процессора. Если загрузка процессора достаточно большая, то подсистему Alarm Processor рекомендуется запускать

в отдельном потоке (ManagerThread=1).

Секция **[AtePath2ObjName]** содержит правила преобразования компонентных адресов переменных в SNMP-адреса.

Правило преобразования состоит из двух частей, разделенные символом «;»: правила преобразования адреса компонента в SNMP-адрес и соответствие имени переменной SNMP-идентификатору.

Пример правила преобразования:

```
{MKD(600).Mod_MCU(3).MCU(1,1);CA(4096);};
```

Правило преобразования, приведенное в примере выше - «{MKD(600).Mod_MCU(3).MCU(1,1);CA(4096);};», определяет преобразование из компонентного адреса «MKD.Mod_MCU.MCU.CA» в SNMP-адрес «600.3.1.1.4096».

Необходимо отметить, что при передаче по сети, к этому SNMP-адресу всегда будет добавляться SNMP-адрес корневого узла дерева переменных, о котором говорилось в пункте 2.2.

Секция **[SNMP]** определяет сетевые параметры протокола SNMP:

- ListenIP — IP-адрес mCore.MKD;
- ListenPort — порт, используемый протоколом SNMP (по умолчанию — 161);
- OwnEnterprise — ветвь в глобальном SNMP-дереве для продуктов «НТЦ Протей» (этот параметр имеет постоянное значение — 1.3.6.1.4.1.20873.100).

В секции **[StandardMib]** определяется список стандартных переменных и их значений.

Формат записи, описывающей стандартную переменную:

```
{<SNMP-адрес переменной>;<тип ответа>;<ответ>;};
```

Секция **[SNMPTrap]** определяет параметры взаимодействия с SNMP-менеджерами. Подсистема Alarm Processor может взаимодействовать с несколькими SNMP-менеджерами одновременно. Для каждого SNMP-менеджера в секции [SNMPTrap] можно определить свои параметры.

Формат записи секции [SNMPTrap]:

```
{<IP-адрес SNMP-менеджера>;<порт SNMP-менеджера>;<фильтр компонентного адреса>;<фильтр типа компонента>;<фильтр компонентного адреса переменной>;};
```

Секция **[Filter]** определяет фильтры по адресу компонентов-источников трапов, по типу компонентов-источников трапов, по компонентному адресу переменной. Эти фильтры «отсеивают» трапы на входе подсистемы Alarm Processor, т.е. между логикой, которая является источником трапов и подсистемой Alarm Processor.

В таблице ниже приведено описание параметров секции [Filter].

Таблица. Параметры секции [Filter].

Имя параметра	Описание
CA_Object	Фильтр по адресу компонента-источника трапа.
CT_Object	Фильтр по типу компонента-источника трапа.
CA_Var	Фильтр по адресу переменной.
TrapIndicator	Фильтр по индикатору трапа.
DynamicIndicator	Фильтр по индикатору динамического объекта.

Пример секции [Filter], в котором разрешены трапы от всех компонентов, имеющие любой тип, адрес переменной может быть любым:

```
CA_Object=".*";
CT_Object=".*";
CA_Var=".*";
```

3.2 Алгоритм формирования идентификатора трапа

В формировании идентификатора трапа участвует информация, содержащаяся в файлах ar.cfg и ar_dictionary.

При возникновении какого-либо события подсистема Alarm Processor получает адрес компонента источника события и переменную, связанную с событием. Далее, используя информацию файлов ar.cfg и ar_dictionary, Подсистема Alarm Processor вычисляет значение идентификатора трапа. Файл ar.cfg предоставляет информацию для вычисления базового значения идентификатора трапа, ar_dictionary — смещения. В условной форме формулу вычисления идентификатора трапа можно записать:

$$\text{trap_id} = \text{ar.cfg} * 1000 + \text{ar_dictionary}$$

Эту запись можно интерпретировать так: значение, найденное в файле ar.cfg умножается на 1000, и к этому результату прибавляется значение, найденное в файле ar_dictionary.

Поиск в файле ar.cfg выполняется путем последовательного просмотра секций: **[SpecificTrapCA_Object]** — поиск по адресу компонента источника события в секции **[SpecificTrapCT_Object]** — поиск по типу компонента источника события в секции, **[SpecificTrapCA_Var]** — поиск по адресу переменной в секции. Поиск прекращается на первом попавшемся совпадении. Это означает, что если, например, в секции **[SpecificTrapCA_Object]** будет найден адрес компонента источника события, то поиск в файле ar.cfg прекращается, далее будет выполняться поиск имени переменной в файле ar_dictionary. И наоборот, если результат поиска в секции **[SpecificTrapCA_Object]** отрицательный, то поиск продолжается в секции **[SpecificTrapCT_Object]** по типу компонента источника события, и далее в секции **[SpecificTrapCA_Var]**, если в **[SpecificTrapCT_Object]** нет искомого типа компонента.

Рассмотрим работу алгоритма на конкретном примере. Допустим в файле ar.cfg представлено следующее содержимое секций [SpecificTrapCT_Object] и [SpecificTrapCA_Var]:

```
[SpecificTrapCA_Var]                # Rules for creating SNMP trap id
{"Calls"; 101;};

[SpecificTrapCT_Object]             # Rules for creating SNMP trap id
{"MKD.Sys"; 1;};
{"MKD.Mod_MCU.MCU"; 2;};
```

Файл ar_dictionary содержит следующие данные:

```
OSTATE =  
{  
    1; SP_Trap = 1;  
    0; SP_Trap = 2;  
    -1; SP_Trap = 3;  
};
```

Предположим, что в подсистему Alarm Processor от логики поступило событие об изменении оперативного состояния (переменная OSTATE) компоненты с адресом «MKD.Sys». Подсистема Alarm Processor начинает поиск в файле ar.cfg, в секции [SpecificTrapCA_Object]. Данная секция в ar.cfg отсутствует. Далее продолжается поиск в секции [SpecificTrapCT_Object]. В данной секции обнаруживается запись («{"MKD.Sys"; 1;}» с искомым типом компонента. Эта запись содержит соответствующий идентификатор - «1». На этом поиск в файле ar.cfg прекращается. Найденный идентификатор «1» умножается на «1000» в результате получаем базовое значение идентификатора - «1000». Далее подсистема Alarm Processor выполняет поиск переменной с именем «OSTATE» в файле ar_dictionary. Поиск завершается с положительным результатом — блок «OSTATE». Блок «OSTATE» в примере выше содержит три записи, каждая из которых состоит из значения переменной и соответствующего идентификатора. Результатом поиска будет идентификатор, соответствующий значению переменной «OSTATE». Допустим переменная «OSTATE» имеет значение «-1», ему в примере соответствует идентификатор «3».

В завершение складываем базовое значение идентификатора («1000») с результатом поиска в файле ar_dictionary - «3», получаем «1003». Это и будет итоговое значение идентификатора трапа.

Примечание. Отрицательному результату поиска в файле ar_dictionary соответствует значение «0», т.е. итоговое значение идентификатора трапа будет совпадать с базовым значением, если взять пример выше - «trap_id = 1000 + 0».

4 Использование SNMP-менеджеров

Протокол SNMP, в силу своей универсальности, позволяет использовать для мониторинга сетевого компонента и его управления любой известный SNMP-менеджер. SNMP-менеджер обычно имеет графический интерфейс пользователя, предоставляющий оператору удобные средства контроля текущего состояния сетевого компонента.

Любой SNMP-менеджер способен взаимодействовать с несколькими сетевыми компонентами одновременно. Для того чтобы SNMP-менеджер начал контролировать какой-либо сетевой компонент (например, mCore.MKD), данный сетевой компонент должен быть зарегистрирован в SNMP-менеджере. Дополнительно для данного компонента должен быть создан mib-файл, описывающий переменные и трапы. mib-файл формируется производителем сетевого компонента (в нашем случае это mCore.MKD), и включается в состав программного обеспечения поставляемого изделия. Все SNMP-менеджеры способны отображать содержимое mib-файла: список переменных и трапов и их свойства.

Контролируемый сетевой компонент в большинстве случаев отображается SNMP-менеджером в виде значка (иконки) с подписью. Возможны и другие способы отображения.

SNMP-менеджер при взаимодействии с mCore.MKD выполняет следующие действия:

- чтение значений переменных по запросу оператора (просмотр переменных);
- прием трапов.

При просмотре значений переменных, SNMP-менеджер обычно отображает их в виде иерархического списка (дерево) или в виде таблицы. Дополнительно в SNMP-менеджере может быть реализована возможность представления значений переменной в виде графика. Данное свойство предоставляет достаточно наглядный способ контроля быстро изменяющихся переменных (например, температуры).

Трапы по своим свойствам отличаются от простых переменных. Во-первых, трапы — это асинхронные события. Во-вторых, трапы имеют приоритет (или, другими словами, уровень важности). В-третьих, для трапа можно определить правила его обработки и фильтрации.

В таблице ниже приведен список приоритетов трапов.

Таблица. Приоритеты трапов

Приоритет	Числовое значение
Critical	1
Severe	2
Major	3
Minor	4
Warning	5
Normal	6
Info	7

SNMP-менеджер отображает информацию, связанную с трапами несколькими способами одновременно. Например, путем изменения цвета иконки сетевого компонента источника трапа, или в виде таблицы, содержащей информацию о пришедших трапах. В зависимости от приоритета трапа, строка в таблице может быть выделена своим цветом. Как правило, таблица с трапами имеет набор фильтров, позволяющие ограничить выводимую информацию по принадлежности к какому-либо сетевому компоненту, по приоритету трапа или по другим свойствам.

Каждый трап в наборе свойств, определенных в mib-файле, имеет правила его обработки. SNMP-менеджер может изменять правила обработки трапа, определенные в mib-файле, или добавлять свои. Обычно изменения хранятся в отдельном файле. Наиболее частыми видами обработки трапа являются: вызов какой-либо программы, отправка email-сообщения, подача звукового сигнала, вывод всплывающего окна с соответствующим сообщением, занесение трапа в базу данных.

5 Описание переменных и трапов mCore.MKD

5.1 Переменные

Пример. Фрагмент файла с переменными МКД.

```
System variables of MKD
{ MKD(600).Sys(2,1,1);CA(4096); };
{ MKD(600).Sys(2,1,1); OSTATE(4097); };
{ MKD(600).Sys(2,1,1); Calls(5000); };
{ MKD(600).Sys(2).Period(3,1,1); CA(4096); };
{ MKD(600).Sys(2).Period(3,1,1); NoCgPN(6000); };
{ MKD(600).Sys(2).Period(3,1,1); Call(6001).Out(1).Count(1); };
{ MKD(600).Sys(2).Period(3,1,1); Call(6001).Out(1).Answer(2); };
{ MKD(600).Sys(2).Period(3,1,1); Call(6001).Out(1).Busy(3); };
{ MKD(600).Sys(2).Period(3,1,1); Call(6001).Out(1).NoAns(4); };
{ MKD(600).Sys(2).Period(3,1,1); Call(6001).Out(1).Rel(5); };
{ MKD(600).Sys(2).Period(3,1,1); Call(6001).Out(1).Max(6); };
{ MKD(600).Sys(2).Period(3,1,1); Call(6001).Out(1).Min(7); };
{ MKD(600).Sys(2).Period(3,1,1); Ratio(6002).Err(1); };
{ MKD(600).Sys(2).Period(3,1,1); Call(6003).In(1).Count(1); };
{ MKD(600).Sys(2).Period(3,1,1); Call(6003).In(1).Fail(2); };
{ MKD(600).Sys(2).Period(3,1,1); Call(6003).Input(2).Err(1); };
{ MKD(600).Sys(2).Period(3,1,1); Call(6003).Step1(3).Err(1); };
{ MKD(600).Sys(2).Period(3,1,1); Call(6003).Step3(4).Err(1); };
{ MKD(600).Sys(2).Period(3,1,1); Call(6003).Step4(5).Err(1); };
{ MKD(600).Sys(2).RTCP(2,1,1); CA(4096); };
{ MKD(600).Sys(2).RTCP(2,1,1); USER(6000); };
{ MKD(600).Sys(2).RTCP(2,1,1); DIRECT(6001); };
{ MKD(600).Sys(2).RTCP(2,1,1); ADDR(6002); };
{ MKD(600).Sys(2).RTCP(2,1,1); FRLOST(6003); };
# Variables of MCU
{ MKD(600).Mod_MCU(3).MCU(1,1);CA(4096); };
{ MKD(600).Mod_MCU(3).MCU(1,1);OSTATE(4097); };
{ MKD(600).Mod_MCU(3).MCU(1,1);IP_addr(6000); };
{ MKD(600).Mod_MCU(3).MCU(1,1);Port(6001); };
{ MKD(600).Mod_MCU(3).MCU(1,1);Version(6002); };
{ MKD(600).Mod_MCU(3).MCU(1,1);Users(6003).Resrvd(1); };
{ MKD(600).Mod_MCU(3).MCU(1,1);Users(6003).Used(2); };
{ MKD(600).Mod_MCU(3).MCU(1,1);Alarm(7000).Version(1); };
# variables of RD_NAS
```

```
{ MKD(600).Mod_RD(4).RD(1,1);CA(4096); };
{ MKD(600).Mod_RD(4).RD(1,1);OSTATE(4097); };
# variables of CDR
{ MKD(600).Mod_CDR(5,1,1);CA(4096); };
{ MKD(600).Mod_CDR(5,1,1);PBX_ID(6000); };
```

В таблице ниже приведено описание Alarm-переменных mCore.MKD.

Таблица. Alarm-переменные mCore.MKD.

Переменная	Описание переменной	Тип и значение	Описание значения
Таблица PROTEI.MKD.Sys-Table			
sys-CA (600.2.1.1.4096)	Компонентный адрес модуля «CPS».	Строка.	Компонентный адрес модуля «CPS».
sys-OSTATE (600.2.1.1.4097)	Оперативное состояние модуля CPS. (траповая переменная, см. трапы 1001, 1002, 1003 в подразделе «Трапы»)	active	Нормальная работа модуля CPS.
		failed	Критический сбой в работе модуля CPS.
		unknown	Оперативное состояние модуля CPS неизвестно.
sys-Calls (600.2.1.1.5000)	Количество вызовов, обрабатываемых в данный момент времени модулем CPS. Информационная переменная.	Положительное целое число.	Информационная переменная, для оценки нагрузки на модуль CPS. Период обновления - ~500 мс.
Sys-Period – таблица переменных счетчиков.			

Переменная	Описание переменной	Тип и значение	Описание значения
Sys-Period-CA (600.2.3,1,1.4096)	<p>CA = MKD.Sys.Period.* , где * - id периода за который отправляется статистика.</p> <p>Сейчас статистика отправляется за 5 секунд, 1 минуту, 10 минут и 1 час. Т.е. CA = MKD.Sys.Period.5, MKD.Sys.Period.60, MKD.Sys.Period.600, MKD.Sys.Period.3600.</p> <p>Примечание: За 5-секундные интервалы статистика выдается только по всему МКД и по всем направлениям, без детализации по классам ошибок и без суммарной нагрузки в секундах за период времени.</p> <p>Статистика в МКД ведется по каждому PBX (PBX.*) и всему MKD(ALL) в целом.</p> <p>Для каждого объекта по которому ведется статистика, она подразделяется по направлениям:</p> <ul style="list-style-type: none"> To_ALL - на все направления; To_INSIDERS - на инсайдеров; DIRECTION.* - на именованные направления(* - имя направления); <p>DirectionsList - определяет список направления для объекта по которому ведется статистика. В объекте ALL - перечислены списки направлений для всех PBX системы.</p> <p>(траповая переменная, трап 4101).</p>		
Sys-Period-NoCgPN (600.2.3,1,1.6000)	Количество поступивших в систему вызовов без CgPN.	Положительное целое число	Категория счетчика. Информационная переменная. Пишется за период для МКД и каждого PBX, не детализируется по направлениям
Sys-Period-Call-Out-Count (600.2.3,1,1.6001.1.1)	Количество попыток вызовов за период времени.	Положительное целое число.	Информационная переменная.
MKD-Sys-Period-Call-Out-Answer (600.2.3,1,1.6001.1.2)	Количество успешных вызовов за период времени (вызовов с ответом вызываемого абонента).	Положительное целое число.	Категория счетчика. Информационная переменная.
MKD-Sys-Period-Call-Out-Busy (600.2.3,1,1.6001.1.3)	Количество вызовов на занятых абонентов за период времени (отбой до ответа с cause=17).	Положительное целое число.	Категория счетчика. Информационная переменная.

Переменная	Описание переменной	Тип и значение	Описание значения
MKD-Sys-Period-Call-Out-NoAns (600.2.3,1,1.6001.1.4)	Количество вызовов без ответа вызываемого абонента за период времени (вызовы, отбитые стороной А, Б или системой до ответа вызываемого абонента, но после получения Alerting со стороны Б).	Положительное целое число.	Категория счетчика. Информационная переменная.
MKD-Sys-Period-Call-Out-Rel (600.2.3,1,1.6001.1.5)	Общее кол-во неуспешных исходящих вызовов.	Положительное целое число.	Категория счетчика. Информационная переменная.
MKD-Sys-Period-Call-Out-Max (600.2.3,1,1.6001.1.6)	Пиковое количество одновременных вызовов за период времени.	Положительное целое число.	Категория счетчика. Информационная переменная.
MKD-Sys-Period-Call-Out-Min (600.2.3,1,1.6001.1.7)	Минимальное количество одновременных вызовов за период времени.	Положительное целое число.	Категория счетчика. Информационная переменная.
MKD-Sys-Period-Ratio-Err (600.2.3,1,1.6002.1)	Отношение неуспешных вызовов ко всему количеству вызовов.	Число.	Категория счетчика. Информационная переменная.
MKD-Sys-Period-Call-In-Count (600.2.3,1,1.6003.1.1)	Количество попыток входящих на МКД вызовов за период времени.	Положительное целое число.	Категория счетчика. Информационная переменная.
MKD-Sys-Period-Call-In-Fail (600.2.3,1,1.6003.1.2)	Общее количество неуспешных входящих вызовов.	Положительное целое число.	Категория счетчика. Информационная переменная.
MKD-Sys-Period-Call-Input-Err (600.2.3,1,1.6003.2.1)	Количество раз набора номера вне плана набора.	Положительное целое число.	Категория счетчика. Информационная переменная.
MKD-Sys-Period-Call-Step1-Err (600.2.3,1,1.6003.3.1)	Количество раз ошибочного выполнения Step1 (внутреннее правило преобразования CgPn/CdPn в формат, требуемый МКД).	Положительное целое число.	Категория счетчика. Информационная переменная.

Переменная	Описание переменной	Тип и значение	Описание значения
MKD-Sys-Period-Call-Step3-Err (600.2.3,1,1.6003.4.1)	Количество раз ошибочного выполнения Step3 (внутреннее правило маршрутизации вызова на один из шлюзов МКД).	Положительное целое число.	Категория счетчика. Информационная переменная.
MKD-Sys-Period-Call-Step4-Err (600.2.3,1,1.6003.5.1)	Количество раз ошибочного выполнения Step4 (внутреннее правило преобразования номеров в необходимый для соединения формат).	Положительное целое число.	Категория счетчика. Информационная переменная.
Sys-RTCP – таблица переменных, получаемых при превышении порога потерь RTP пакетов у абонента на прием, за отчетный период.			
MKD-Sys-RTCP-CA (600.2.2,1,1.4096)	Адрес компоненты. Траповая переменная, трап посылается при превышении порога потерь RTP пакетов у абонента на прием. (Трап 5101).	Строка.	Компонентный адрес MKD_RTCP.
MKD-Sys-RTCP-USER (600.2.2,1,1.6000)	Номер пользователя.	Набор чисел.	Внешний номер абонента, который перестал принимать пакеты
MKD-Sys-RTCP-DIRECT (600.2.2,1,1.6001)	Адрес направления.	Строка.	Направление, где находится абонент
MKD-Sys-RTCP-ADDR (600.2.2,1,1.6002)	Адрес абонента.	Строка	Адрес абонента.03
MKD-Sys-RTCP-FRLOST (600.2.2,1,1.6003)	Доля потерь за отчетный период.	Набор чисел.	Доля потерь - это отношение потерянных пакетов к ожидаемым, умноженное на 256.
Таблица PROTEI.MKD.MCU-Table			
mcsu-CA (600.3.1.1.4096)	Компонентный адрес модуля MCU, подключенного к данному CPS.	Строка.	Значение – MKD.Mod_MCU.MCU.*, где * - номер MCU.

Переменная	Описание переменной	Тип и значение	Описание значения
MKD-Mod_MCU-MCU-OSTATE (600.3.1.1.4097)	Переменная показывает совместимость версии МКД и MCU. Если версии не совместимы, отправляется трап со значением 2.	1	Норма.
		2	Версии несовместимы.
mcu-Connection	Состояние соединения модуля MCU с модулем CPS (траповая переменная, см. трапы 2001, 2002, 2003 в подразделе «Трапы»).	active	Соединение активно.
		failed	Соединение не установлено.
		unknown	Данное значение возможно в начале загрузки ПО, когда переменная еще не инициализирована. Дождитесь окончания загрузки ПО.
mcu-IP_addr (600.3.1.1.6000)	IP-адрес модуля MCU. Информационная переменная	Строка.	
mcu-Port (600.3.1.1.6001)	Порт, используемый для соединения с MCU. Информационная переменная.	Положительное целое число.	
mcu-Users-Resrvd (600.3.1.1.6003.1)	Количество зарезервированных ресурсов (портов для RTP-каналов и т.д.). Информационная переменная.	Положительное целое число.	Для неактивных соединений с MCU переменная будет иметь значение 0.
mcu-Users-Used (600.3.1.1.6003.2)	Количество используемых ресурсов (портов для RTP-каналов и т.д.). Информационная переменная.	Положительное целое число.	Для неактивных соединений с MCU переменная будет иметь значение 0.

Переменная	Описание переменной	Тип и значение	Описание значения
mcu-Alarm-Version (600.3.1.1.7000.1)	Совместимость версий модулей CPS и MCU. Если версия MCU устаревшая/несовместима с модулем CPS, то данный модуль MCU не будет использоваться CPS. Проверка выполняется при значении параметра StrictVersionCheck = 1 (параметр содержится в конфигурационном файле /usr/protei/MKD/MCU/config/config.cfg). (траповая переменная, см. трапы 2004, 2005 в подразделе «Трапы»).	compatible	Версии совместимы.
		not_compatible	Версии не совместимы. Обновите версию MCU.
mcu-Version (600.3.1.1.6002)	Версия MCU. Информационная переменная.	Строка.	Для неактивных соединений с модулем MCU переменная имеет значение - «0.0.0.0».
Таблица переменных RD_NAS			
RD_NAS-CA (600.4.1.1.4096)	Компонентный адрес модуля интерфейса с RADIUS.	Строка.	
RD_NAS-OSTATE (600.4.1.1.4097)	Оперативное состояние соединения mCore.MKD с RADIUS (траповая переменная, см. трапы 3001, 3002, 3003 в подразделе «Трапы»).	active	Соединение активно.
		failed	Соединение не установлено. Проверить что случилось с RADIUS сервером или RD_NAS. Вызовы требующие авторизации на RADIUS-сервере перестанут проходить.
		unknown	Данное значение возможно в начале загрузки ПО, когда переменная еще не инициализирована. Дождитесь окончания загрузки ПО.
Таблица переменных CDR			

Переменная	Описание переменной	Тип и значение	Описание значения
MKD-Mod_CDR-CA (600.5,1,1.4097)	Адрес переменной Траповая переменная, выбрасывается при некорректной обработке скрипта преобразования номеров CgPN, CdPN перед записью в CDR. (трап 6101).	Адрес компоненты MKD-Mod_CDR.	
MKD-Mod_CDR-PBX_ID (600.5,1,1.6000)	Идентификатор PBX'а для которого вызывался скрипт.		

5.2 Трапы

В таблице ниже приведено описание SNMP-трапов mCore.MKD.

Таблица. SNMP-трапы mCore.MKD.

№ трапа	Название трапа	Текстовое сообщение (английский)	Приоритет и действия SNMPc при получении трапа	Описание
1001	trapSystem-OSTATE-Active	MKD is active	Нормальный (normal), запись в журнал.	Трап посылается при загрузке ПО mCore.MKD. На данный трап следует обратить внимание, если оператор не производил ручного старта или перезапуска МКД. В этом случае следует по журналам МКД выяснить причину перезапуска ПО.
1002	trapSystem-	MKD is failed	Критичный (critical)	Трап посылается при критическом сбое в ПО.

№ трапа	Название трапа	Текстовое сообщение (английский)	Приоритет и действия SNMPc при получении трапа	Описание
	OSTATE-Failed		запись в журнал, звуковой сигнал, окно аварийного сообщения.	mCore.MKD. Необходимо проанализировать причины сбоя по журналам info.log, trace.log, warning.log. При невозможности продиагностировать проблему необходимо обратиться в службу технической поддержки, приложив все имеющиеся журналы ПРОТЕЙ-МКД
1003	trapSystem-OSTATE-Unknown	Operative state of MKD is unknown	Информационный (info), запись в журнал.	Трап может прийти во время загрузки ПО mCore.MKD. Дождаться полной загрузки ПО ПРОТЕЙ-МКД и прихода трапа №1001 или №1002
1101	trapSystem-Calls-CurrentValue	Current calls number processing by MKD - * («*» - текущее количество вызовов, обслуживаемых CPS).	Информационный (info), запись в журнал.	Трапе содержит количество вызовов, обслуживаемых mCore.MKD в текущий момент.
2001	trapMCU-Connection-Active	Connection between MKD and module MCU with address MKD.Mod_MCU.MCU.* is active («*» - номер модуля MCU).	Нормальный (normal), запись в журнал.	Трап посылается при нормальном установлении соединения с модулем MCU.
2002	trapMCU-Connection-Failed	Connection between MKD and module MCU with address MKD.Mod_MCU.MCU.* is failed («*» - номер модуля MCU).	Важный (major), запись в журнал, звуковой сигнал, окно аварийного сообщения.	Трап посылается при потере соединения с модулем MCU. Необходимо проверить совместимость версий МКД (модуля CPS) и модуля MCU с помощью переменной mcu-AlarmVersion и доступность модуля CPS с модуля MCU (например, командой ping).
2003	trapMCU-Connection-Unknown	State of connection between MKD and module MCU with	Информационный (info), запись в журнал.	Происходит загрузка ПО оборудования и переменная еще не инициализирована. Дождитесь окончания загрузки ПО.

№ трапа	Название трапа	Текстовое сообщение (английский)	Приоритет и действия SNMPc при получении трапа	Описание
		address MKD.Mod_MCU.MCU.* is unknown («*» - номер модуля MCU).		
2004	trapMCU-AlarmVersion-NotCompatible	Version of module MCU with address MKD.Mod_MCU.MCU.* is not compatible with version of MKD («*» - номер модуля MCU).	Важный (major), запись в журнал, звуковой сигнал, окно аварийного сообщения.	Проверка версии MCU завершилась неуспешно (несовместимая версия модуля MCU). Замените версию модуля MCU на более новую, выполнить рестарт ПО.
2005	trapMCU-AlarmVersion-Compatible	Version of module MCU with address MKD.Mod_MCU.MCU.* is compatible with version of MKD («*» - номер модуля MCU).	Нормальный (normal), запись в журнал.	Проверка версии MCU завершилась успешно.
3002	trapRD-Connection-Failed	Connection between MKD and RD with address \$2 is failed	Важный (major), запись в журнал, звуковой сигнал, окно аварийного сообщения.	Трап посылается при потере соединения с RADIUS. Проверить что случилось с RADIUS сервером или RD_NAS. Вызовы требующие авторизации на RADIUS-сервере перестанут проходить.
3001	trapRD-Connection-Active	Connection between MKD and RD with address \$2 is active	Нормальный (normal), запись в журнал.	Трап посылается при нормальном установлении соединения с RADIUS.
3003	trapRD-Connection-Unknown	State of connection between MKD and RD with address \$2 is unknown	Информационный (info), запись в журнал.	Происходит загрузка ПО оборудования и переменная еще не инициализирована. Дождитесь окончания загрузки ПО.
4101	trapPer-Statistic	Statistic from mkd	Нормальный (normal),	Трапы по статистическим параметрам МКД -

№ трапа	Название трапа	Текстовое сообщение (английский)	Приоритет и действия SNMPc при получении трапа	Описание
		address \$1	запись в журнал.	Call.Out.Count, Call.Out.Answer, Call.Out.Busy, Call.Out.NoAns, Call.Out.Rel, Call.Out.Max, Call.Out.Min,Ratio.Err, NoCgPN, Call.In.Count, Call.In.Fail,Call.Input.Err,Call.Step1.Err,Call.Step3.Err, Call.Step4.Err.
5101	trapRTCP	Tcp: very much package was lost. FracLost - \$2, User - \$3, Dir - \$4, Addr - \$	Важный (major), запись в журнал.	Трап посылается при превышении порога потерь RTP-пакетов у абонента на прием.
6101	trapCDR-Error-Script	Script for cdr' was finished with error, PBX - \$2	Важный (major), запись в журнал, звуковой сигнал, окно аварийного сообщения.	Трап посылается при некорректной отработке скрипта преобразования номеров CgPN, CdPN перед записью в CDR.

Пример. Фрагмент миб-файла с трапами МКД.

```

trapSystem-OSTATE-Failed TRAP-TYPE
    ENTERPRISE mkd
    VARIABLES { sys-CA, sys-OSTATE }
    --&ACTIONS { log, critical, alarm, sound }
    --&CLEARS { 1001, 1003 }
    --&MSG "mkd is failed"
    DESCRIPTION "mkd failed"
    ::= 1002

trapSystem-OSTATE-Active TRAP-TYPE
    ENTERPRISE mkd
    VARIABLES { sys-CA, sys-OSTATE }
    --&ACTIONS { log, normal }
    --&CLEARS { 1002, 1003 }
    --&MSG "mkd is active"
    DESCRIPTION "mkd active"
    ::= 1001

trapSystem-OSTATE-Unknown TRAP-TYPE
    ENTERPRISE mkd

```

```

VARIABLES { sys-CA, sys-OSTATE }
--&ACTIONS { log, info }
--&CLEARS { 1001, 1002 }
--&MMSG "Operative state of mkd is unknown"
DESCRIPTION "Operative state of mkd is unknown"
::= 1003

trapSystem-Calls-CurrentValue TRAP-TYPE
ENTERPRISE mkd
VARIABLES { sys-CA, sys-Calls }
--&ACTIONS { log, info }
--&MMSG "Current calls number processing by mkd - $2"
DESCRIPTION "mkd calls number"
::= 1101

trapMCU-Connection-Failed TRAP-TYPE
ENTERPRISE mkd
VARIABLES { mcu-CA, mcu-Connection }
--&ACTIONS { log, major, alarm, sound }
--&CLEARS { 2001, 2003 }
--&MMSG "Connection between mkd and module mcu with address $1 is failed"
DESCRIPTION "mcu failed"
::= 2002

trapMCU-Connection-Active TRAP-TYPE
ENTERPRISE mkd
VARIABLES { mcu-CA, mcu-Connection }
--&ACTIONS { log, normal }
--&CLEARS { 2002, 2003 }
--&MMSG "Connection between mkd and module mcu with address $1 is active"
DESCRIPTION "mcu active"
::= 2001

trapMCU-Connection-Unknown TRAP-TYPE
ENTERPRISE mkd
VARIABLES { mcu-CA, mcu-Connection }
--&ACTIONS { log, info }
--&CLEARS { 2001, 2002 }
--&MMSG "State of connection between mkd and module mcu with address $1 is unknown"
DESCRIPTION "Operative state of mcu is unknown"
::= 2003

trapMCU-AlarmVersion-Compatible TRAP-TYPE
ENTERPRISE mkd
VARIABLES { mcu-CA, mcu-AlarmVersion }

```

```

--&ACTIONS { log, normal }
--&CLEARS { 2004 }
--&MSG "Version of module mcu with address $1 is compatible with version of mkd"
DESCRIPTION "Version of mcu is compatible with version of mkd"
::= 2005

trapMCU-AlarmVersion-NotCompatible TRAP-TYPE
ENTERPRISE mkd
VARIABLES { mcu-CA, mcu-AlarmVersion }
--&ACTIONS { log, major, alarm, sound }
--&CLEARS { 2005 }
--&MSG "Version of module mcu with address $1 is not compatible with version of mkd"
DESCRIPTION "Version of mcu is not compatible with version of mkd"
::= 2004

trapRD-Connection-Failed TRAP-TYPE
ENTERPRISE mkd
VARIABLES { rd-CA, rd-Connection }
--&ACTIONS { log, major, alarm, sound }
--&CLEARS { 3001, 3003 }
--&MSG "Connection between mkd and rd with address $1 is failed"
DESCRIPTION "rd failed"
::= 3002

trapRD-Connection-Active TRAP-TYPE
ENTERPRISE mkd
VARIABLES { rd-CA, rd-Connection }
--&ACTIONS { log, normal }
--&CLEARS { 3002, 3003 }
--&MSG "Connection between mkd and rd with address $1 is active"
DESCRIPTION "rd active"
::= 3001

trapRD-Connection-Unknown TRAP-TYPE
ENTERPRISE mkd
VARIABLES { rd-CA, rd-Connection }
--&ACTIONS { log, info }
--&CLEARS { 3001, 3002 }
--&MSG "State of connection between mkd and rd with address $1 is unknown"
DESCRIPTION "Operative state of rd is unknown"
::= 3003

trapPer-Statistic TRAP-TYPE
ENTERPRISE mkd

```

```
VARIABLES { per-CA, per-NoCgPN, per-OutCoun, per-OutAnsw, per-OutBusy, per-OutNoAns, per-OutRel, per-OutMax, per-OutMin, per-Ratio, per-
InCount, per-InFail, per-InputErr, per-st1, per-st3, per-st4}
--&ACTIONS { log, normal }
--&MMSG "Statistic from mkd address $1"
DESCRIPTION "Receive statistic from mkd"
::= 4101

trapRTCP-veryMuchLost TRAP-TYPE
ENTERPRISE mkd
VARIABLES { rtcp-CA, rtcp-FRLOST, rtcp-USER, rtcp-DIRECT, rtcp-ADDR }
--&ACTIONS { log, major}
--&MMSG "rtcp: very much package was lost. FracLost - $2, User - $3, Dir - $4, Addr - $5"
DESCRIPTION "rtcp: very much package was lost"
::= 5101

trapCDR-Error-Script TRAP-TYPE
ENTERPRISE mkd
VARIABLES { cdr-CA, cdr-PBX-ID }
--&ACTIONS { log, major, alarm, sound }
--&MMSG "Script for cdr's was finished with error, PBX - $2"
DESCRIPTION "cdr-script failed"
::= 6101
```

6 Приложение

Пример реального файла конфигурации ar.cfg.

```
[General] # General parameters of AP
ApplicationAddress=MKD.212; # Not used
MaxConnectionCount=10; # Maximum number of connected SNMP managers
ManagerThread=1; # If =1 AP has its own process, if =0 AP works in main
process of MKD
CyclicTreeWalk=1; # Its recomended not to change this parameter

[Dynamic] # This section describes list variables. If variable
caVar has strValue traps with it are send not only for change but regulary
# Format {caVar;strValue;};

[AtePath2ObjName] # Rules of transformation variables of Component-Address
Model to SNMP tree
# Format {ctObject;caVar;};

# System variables of MKD
{ MKD(600).Sys(2,1,1); CA(4096); };
{ MKD(600).Sys(2,1,1); OSTATE(4097); };
{ MKD(600).Sys(2,1,1); Calls(5000); };

# Variables of MCU
{ MKD(600).Mod_MCU(3).MCU(1,1);CA(4096); };
{ MKD(600).Mod_MCU(3).MCU(1,1);OSTATE(4097); };
{ MKD(600).Mod_MCU(3).MCU(1,1);IP_addr(6000); };
{ MKD(600).Mod_MCU(3).MCU(1,1);Port(6001); };
{ MKD(600).Mod_MCU(3).MCU(1,1);Version(6002); };
{ MKD(600).Mod_MCU(3).MCU(1,1);Users(6003).Resrvd(1); };
{ MKD(600).Mod_MCU(3).MCU(1,1);Users(6003).Used(2); };
{ MKD(600).Mod_MCU(3).MCU(1,1);Alarm(7000).Version(1); };

# variables of RD_NAS
{ MKD(600).Mod_RD(4).RD(1,1);CA(4096); };
{ MKD(600).Mod_RD(4).RD(1,1);OSTATE(4097); };

[SNMP]
ListenIP = 192.168.100.212; # IP-address of MKD to work with SNMP
ListenPort = 161; # Port for sending and receiving SNMP messages
OwnEnterprise = 1.3.6.1.4.1.20873.600; # Branch in SNMP tree for Protei+product

[StandardMib] # List of standart SNMP variables and its value
# Format {request;answer_type;answer;};

# SysDescr
{1.3.6.1.2.1.1.1.0;STRING;"MKD";};
# SysObjectID
{1.3.6.1.2.1.1.2.0;OBJECT_ID;1.3.6.1.4.1.20873;};

[AtePath2Oid] # Not used
# Format {ctObject;caVar;ObjectID;};

[SNMPTrap] # List of IP-addresses to send traps with some filters
# Format {SNMP_ManagerIP;SNMP_ManagerPort;caObjectFilter;ctObjectFilter;caVarFilter;};
{"192.168.108.111";162;};

[Filter] # Filters between MKD and AP
# filters send to AP_Agent
```

```
#CA_Object=".*";
#CT_Object=".*";
#CA_Var=".*";
#TrapIndicator=-1;
#DynamicIndicator=-1;

[Logs]                                # Parameters of logging
TreeTimerPeriod=300000;                # Period of logging alarm tree

[Test]
Freq=1700000;

[SpecificTrapCA_Var]                   # Rules for creating SNMP trap id
{"Calls"; 101;};

[SpecificTrapCT_Object]                # Rules for creating SNMP trap id
{"MKD.Sys"; 1;};
{"MKD.Mod_MCU.MCU"; 2;};
{"MKD.Mod_RD.RD"; 3;};
```