Описание технологических решений оптимальных методов базового процесса

Содержание

Введение

В этом документе описываются основные понятия и процедуры, применяемые в отношении сетей с высокой доступностью. К их числу относятся решающие условия определения базовых и пороговых параметров сети, помогающих в оценке результативности. Также отмечаются важные аспекты процессов определения базовых и пороговых показателей и реализации, выведенные из практического опыта рабочей группы Cisco по службам высокой доступности (HAS).

Этот документ берет вас постепенно посредством процесса эталонного тестирования. Некоторая система управления текущей сети (NMS), продукты могут помочь автоматизировать этот процесс, однако, процесс эталонного тестирования, остается тем

же, используете ли вы автоматизированные или ручные инструменты. При использовании этих продуктов NMS необходимо отрегулировать параметры порога по умолчанию для среды уникальной сети. Важно иметь процесс для интеллектуального выбора тех порогов так, чтобы они были значимы и корректны.

Срок

Что такое базовый анализ?

Срок является процессом для изучения сети через определенные промежутки времени, чтобы гарантировать, что сеть работает, как разработано. Это - больше, чем одиночный отчёт, детализирующий состояние сети в определенный момент вовремя. Следующим базовый процесс можно получить следующую информацию:

- Полезная информация усиления на состоянии программного и аппаратного обеспечения
- Определите текущие ресурсы использования сети
- Сделайте точные решения о порогах сетевого сигнала
- Определите проблемы текущей сети
- Предскажите дальнейшие проблемы

Другой способ посмотреть на срок проиллюстрирован в следующей схеме.

Красная линия (точка разрыва сети) является точкой, в которой сеть будет разорвана; она определяется, исходя из знаний о функционировании аппаратного и программного обнспечения. Зеленая линия (сетевая нагрузка) представляет фактическое изменение нагрузки на сеть по мере добавления новых приложений и других подобных факторов.

Цель срока состоит в том, чтобы определить:

- Где ваша сеть находится на зеленой линии
- Как быстро увеличивается сетевая нагрузка
- Надо надеяться, предскажите, в каком моменте времени пересекутся эти два

Путем выполнения срока регулярно, можно узнать текущее состояние *и* экстраполировать, когда сбои произойдут и подготовят к ним заранее. Это позволяет принимать более компетентные решения о том, когда, куда и как расходовать бюджетные деньги на обновления сети.

Зачем нужен базовый анализ?

Базовый процесс помогает вам определять и должным образом планировать проблемы ограничения критического ресурса в сети. Эти проблемы могут быть описаны как ресурсы платы данных или ресурсы уровня управления. Ресурсы уровня управления уникальны для определенной платформы и модулей в устройстве и могут повлияться многими проблемами включая:

- Использование данных
- Опции активированы
- Проект сети

Ресурсы уровня управления включают параметры, такие как:

- Использование CPU
- Загруженность памяти
- Использование буфера

На ресурсы платы данных влияют только тип и объем трафика и включают использование соединения и использование объединительной платы. Использованием ресурса эталонного тестирования для критических областей можно избежать серьезных проблем производительности, или хуже, разрушение сети.

С введением чувствительных к задержкам приложений, таких как аудио и видео, базовый анализ теперь еще более важен. Традиционный Протокол управления передачей / Протокол Интернета (TCP/IP) приложения являются прощающими и обеспечивают определенную величину задержки. Голос и видео являются основанным Протоколом UDP и не обеспечивают повторные передачи или перегрузку сети.

Из-за нового соединения приложений, эталонное тестирование помогает вам понимать и уровень управления и проблемы использования ресурса платы данных и заранее планировать изменения и обновления гарантировать дальнейший успех.

Сети передачи данных были вокруг много лет. До недавнего времени поддержание сети в работающем состоянии было простительным, с некоторым допуском для ошибки. В связи с расширением сферы применения приложений, чувствительных ко времени ожидания, таких как VoIP, управление сетью становится все более сложной задачей, требующей предельной точности. Чтобы быть более точным и дать администратору сети прочную основу, на которую можно управлять сетью, важно иметь некоторую идею того, как работает сеть. Для этого необходимо проделать процесс, называемый Baseline (Основание).

Цели базового уровня

Цель срока к:

1. Определите текущий статус сетевых устройств
2. Сравните тот статус со стандартной производительностью рекомендации
3. Установите пределы для получения предупреждения, когда состояние выходит за рамки этих инструкций

Из-за большого количества данных и периода времени это берет для анализа данных, необходимо сначала ограничить область срока, чтобы упростить изучать процесс. Наиболее логичным и рекомендуемым местом, с которого необходимо начинать анализ, является ядро сети. Эта часть сети является обычно самой маленькой и требует большей части устойчивости.

Ради простоты этот документ объясняет как сроку одна очень важная Информационная база управления Простоя протокол управления сетью (SNMP MIB): cpmCPUTotal5min. cpmCPUTotal5min является пятиминутным затухающим средним числом процессора (CPU) маршрутизатора Cisco и является индикатором производительности уровня управления. Базовый анализ проводится на Cisco 7000 series router.

Усвоив процесс, можно применять его к любым доступным данным в огромной базе данных SNMP, доступной на большинстве устройств Cisco, например:

- Использование Цифровой сети с интеграцией услуг (ISDN)
- Потеря ячеек Асинхронного режима передачи (ATM)

- Память свободной системы

Основная базовая блок-схема

Следующая блок-схема демонстрирует основные этапы базового процесса ядра. В то время как продукты и программные средства доступны для выполнения некоторых из этих шагов для вас, они имеют тенденцию иметь разрывы в гибкости или простоте использования. Даже если вы планируете использовать программные средства системы управления сетью (NMS) для выполнения эталонного тестирования, это - все еще хорошее упражнение в изучении процесса и понимании, как действительно работает сеть. Этот процесс может также раскрыть хитрости работы некоторых средств системы управления сетью, так как большинство таких средств, по сути, выполняет одинаковые задачи.

Основная процедура

Шаг 1: Скомпилируйте аппаратные средства, программное обеспечение и материально-технические ресурсы конфигурации

Чрезвычайно важно, чтобы вы скомпилировали материально-технические ресурсы аппаратных средств, программное обеспечение и конфигурацию по нескольким причинам. Во-первых, SNMP MIB Cisco являются, в некоторых случаях, определенными для Cisco IOS Release, который вы выполняете. Некоторые объекты MIB заменяются новыми или иногда полностью удаляются. Аппаратный инструментарий имеет наибольшее значение после сбора информации, поскольку пороги, которые необходимо установить после первоначальных базовых значений, часто зависят от типа CPU, объема памяти и других параметров устройств Cisco. Инвентаризация конфигурации также важно для того, чтобы убедиться, что вам известны текущие конфигурации: Можно хотеть изменить конфигурации устройства после срока для настройки буферов и так далее.

Наиболее эффективным способом сделать эту часть базовой архитектуры сети Cisco является использование основных инструментов диспетчера ресурсов CiscoWorks2000 (инструментов). Если это программное обеспечение установлено правильно в сети, Основы должны иметь текущие материально-технические ресурсы всех устройств в ее базе данных. Достаточно просто просмотреть инвентаризационные сведения, чтобы определить наличие каких-либо проблем.

Приведенная ниже таблица - это пример отчета Cisco Router Class software inventory report, экспортированного из Essentials и отредактированного в Microsoft Excel. Из этих материально-технических ресурсов заметьте, что необходимо использовать данные SNMP MIB, и Идентификаторы объекта (OID) нашли в 12.0x и 12.1x Cisco IOS Release.

Device Name	Тип маршрутизатора	Version	Версия программного обеспечения
field-2500a.embu-mlab. cisco . com	Cisco 2511	M	12.1 (1)
qdm-7200.embu-mlab. cisco . com	Cisco 7204	B	12.1 (1) E

voip-3640.embu-mlab. cisco . com	Cisco 3640	0x00	12.0 (3c)
wan-1700a.embu-mlab. cisco . com	Cisco 1720	0x101	12.1 (4)
wan-2500a.embu-mlab. cisco . com	Cisco 2514	L	12.0 (1)
wan-3600a.embu-mlab. cisco . com	Cisco 3640	0x00	12.1 (3)
wan-7200a.embu-mlab. cisco . com	Cisco 7204	B	12.1 (1) E
172.16.71.80	Cisco 7204	B	12.0 (5T)

Если Основы не установлены в сети, можно использовать **snmpwalk** программного средства командной строки UNIX от рабочей станции UNIX для обнаружения версии IOS. Это показано на следующем примере. Если вы не уверены, как эта команда работает, введите **man snmpwalk** в командной строке UNIX для получения дополнительной информации. Версия IOS важна при выборе базовых MIB OID, так как объекты MIB зависят от IOS. Также заметьте, что путем знания типа маршрутизатора, можно позже сделать определения относительно того, чем пороги должны быть для ЦП, буферов, и так далее.

```
nsahpov6% snmpwalk -v1 -c private 172.16.71.80 system
system.sysDescr.0 : DISPLAY STRING- (ascii): Cisco Internetwork Operating System Software
IOS (tm) 7200 Software (C7200-JS-M), Version 12.0(5)T, RELEASE SOFTWARE (fc1)
Copyright (c) 1986-2001 by cisco Systems, Inc.
Compiled Fri 23-Jul-2001 23:02 by kpma
system.sysObjectID.0 : OBJECT IDENTIFIER:
.iso.org.dod.internet.private.enterprises.cisco.ciscoProducts.cisco7204
```

Шаг 2: Подтвердите, что SNMP MIB поддерживается в маршрутизаторе

Теперь, когда у вас есть материально-технические ресурсы устройства, которое вы хотите опросить для вашего срока, можно начать выбирать определенные OID, которые вы хотите опросить. Это сохраняет большое расстройство, если вы проверяете, загодя, что данные, которые вы хотите, фактически там. Объект cpmCPUTotal5min MIB находится в CISCO-PROCESS-MIB.

Чтобы определить OID для опроса необходима таблица преобразования, доступная на веб-узле Cisco's CCO. Для доступа к этому веб-сайту через браузер зайдите на страницу Cisco MIB и откройте ссылку OID.

Для доступа к этому веб-узлу с сервера FTP наберите ftp://ftp.cisco.com/pub/mibs/oid/. От этого узла можно загрузить определенный MIB, который декодировался и сортировался номерами OID.

Следующий пример извлечен из таблицы CISCO-PROCESS-MIB.oid. **В данном примере значение OID для cpmCPUTotal5min MIB - .1.3.6.1.4.1.9.9.109.1.1.1.1.5.**

Примечание: Не забывайте добавлять а"." к началу OID, или вы получите ошибку, когда вы попытаетесь опросить его. Также для создания экземпляра OID необходимо добавить ".1" к его окончанию. Это говорит устройству экземпляр OID, который вы ищете. В некоторых случаях OID имеют несколько экземпляров определенного типа данных, такой как тогда, когда маршрутизатор имеет множественные ЦПУ.

```
ftp://ftp.cisco.com/pub/mibs/oid/CISCO-PROCESS-MIB.oid
### THIS FILE WAS GENERATED BY MIB2SCHEMA
"org" "1.3"
"dod" "1.3.6"
"internet" "1.3.6.1"
"directory" "1.3.6.1.1"
"mgmt" "1.3.6.1.2"
"experimental" "1.3.6.1.3"
"private" "1.3.6.1.4"
"enterprises" "1.3.6.1.4.1"
"cisco" "1.3.6.1.4.1.9"
"ciscoMgmt" "1.3.6.1.4.1.9.9"
"ciscoProcessMIB" "1.3.6.1.4.1.9.9.109"
"ciscoProcessMIBObjects" "1.3.6.1.4.1.9.9.109.1"
"ciscoProcessMIBNotifications" "1.3.6.1.4.1.9.9.109.2"
"ciscoProcessMIBConformance" "1.3.6.1.4.1.9.9.109.3"
"cpmCPU" "1.3.6.1.4.1.9.9.109.1.1"
"cpmProcess" "1.3.6.1.4.1.9.9.109.1.2"
"cpmCPUTotalTable" "1.3.6.1.4.1.9.9.109.1.1.1"
"cpmCPUTotalEntry" "1.3.6.1.4.1.9.9.109.1.1.1.1"
"cpmCPUTotalIndex" "1.3.6.1.4.1.9.9.109.1.1.1.1.1"
"cpmCPUTotalPhysicalIndex" "1.3.6.1.4.1.9.9.109.1.1.1.1.2"
"cpmCPUTotal5sec" "1.3.6.1.4.1.9.9.109.1.1.1.1.3"
"cpmCPUTotal1min" "1.3.6.1.4.1.9.9.109.1.1.1.1.4"
"cpmCPUTotal5min" "1.3.6.1.4.1.9.9.109.1.1.1.1.5"
```

Существует два обычных способа для опроса OID MIB, чтобы удостовериться, что это доступно и функционирует. Это - хорошая идея сделать это перед началом объемного сбора данных так, чтобы вы не напрасно тратили время, опрашивая что-то, что не является там, и закончите с пустой базой данных. Один способ сделать это должно использовать обходчика базы управляющих данных (MIB) от вашей платформы NMS, такой как Диспетчер узлов сети HP OpenView (NNM) или CiscoWorks Windows, и ввести OID, который вы хотите проверить.

Пример из обходчика HP OpenView SNMP MIB.

Другой простой способ для опроса OID MIB должен использовать **snmpwalk** команды UNIX как показано в следующем примере.

```
nsahpov6% cd /opt/OV/bin
nsahpov6% snmpwalk -v1 -c private 172.16.71.80 .1.3.6.1.4.1.9.9.109.1.1.1.1.5.1

cisco.ciscoMgmt.ciscoProcessMIB.ciscoProcessMIBObjects.cpmCPU.cpmCPUTotalTable.cpmCPUTotalEntry.
cpmCPUTotal5min.1 : Gauge32: 0
```

В обоих примерах MIB возвратил значение 0, означая, что для того цикла опроса ЦП составил в среднем 0 процентов загрузки. При наличии затруднений, заставляя устройство ответить корректными данными попытайтесь пропинговать устройство и обратиться к устройству посредством Telnet. При тихом наличии проблемы проверьте конфигурацию SNMP и Строки имени и пароля SNMP. Вы, возможно, должны найти, что альтернативный MIB или другая версия IOS делают эту работу.

Шаг 3: Опрос и регистрация определенных объектов SNMP MIB с маршрутизатора

Существует несколько способов опросить объекты MIB и сделать запись выходных данных. Стандартные продукты, условно - бесплатные продукты, сценарии и программные средства поставщика доступны. Все программные средства фронтэнда используют процесс **SNMP get** для получения информации. Основное отличие в гибкости конфигурации и способе

записи данных в базу данных. Снова, посмотрите на процессор MIB, чтобы видеть, как работают эти различные методы.

Теперь, когда вы знаете, что OID поддерживается в маршрутизаторе, необходимо решить, как часто опросить его и как сделать запись его. Cisco рекомендует, чтобы MIB ЦП был опрошен в пятиминутных интервалах. Более низкий интервал увеличил бы нагрузку сети или устройство, и так как значение MIB является средним значением из пяти минут так или иначе, не было бы полезно опросить его чаще, чем усредненное значение. Также обычно рекомендуется, чтобы опрос по исходному уровню имел, по крайней мере, двухнедельный период так, чтобы можно было проанализировать по крайней мере два еженедельных цикла активности в сети.

На следующих экранах показано, как добавлять объекты MIB с помощью HP OpenView Network Node Manager 6.1. От основного экрана выберите **Options> Data Collection и Thresholds**.

Затем выберите **Edit> Add> MIB Objects**.

Из меню добавьте, что OID натягивает и нажимает **Apply**. Теперь объект MIB добавлен к платформе HP OpenView и может быть опрошен.

Затем необходимо сообщить HP OpenView, какой маршрутизатор опрашивать для этого OID.

Из меню Data Collection выберите **Edit> Add> MIB Collections**.

В поле Source введите имя Domain Naming System (DNS) или IP-адрес маршрутизатора, который будет опрошен.

В списке Set Collection Mode выберите Store, No Thresholds.

Установите Интервал опроса в **5 м** для пятиминутных интервалов.

Щелкните "Применить".

Необходимо выбрать **File> Save** для изменений для взятия влияния.

Чтобы проверить, что набор установлен должным образом, выделите линию сводки набора для маршрутизатора и выберите **Actions> Test SNMP**. Это проведет проверку правильности строки имени и пароля, а также опрос всех экземпляров идентификатора объекта.

Нажмите **Close** и позвольте набору, выполненному в течение недели. В конце недельного периода извлеките данные для анализа.

Легче анализировать данные, если вы выгрузили их в файл ASCII и импортировали в электронную таблицу, такую как Microsoft Excel. **Чтобы сделать это с HP OpenView NNM, вы можете использовать инструмент командной строки snmpColDump.** Каждый набор настроил записи к файлу в/var/opt/OV/share/databases/snmpCollect/каталоге.

Извлеките данные к файлу ASCII, названному **testfile** со следующей командой:

```
snmpColDump /var/opt/OV/share/databases/snmpCollect/cpmCPUTotal5min.1 > testfile
```

Примечание: cpmCPUTotal5min.1 является файлом базы данных, который создал NNM HP

OpenView, когда начался Опрос OID.

Созданный тестовый файл будет похож на приведенный пример.

```
03/01/2001 14:09:10 nsa-gw.cisco.com 1
03/01/2001 14:14:10 nsa-gw.cisco.com 1
03/01/2001 14:19:10 nsa-gw.cisco.com 1
03/01/2001 14:24:10 nsa-gw.cisco.com 1
03/01/2001 14:29:10 nsa-gw.cisco.com 1
03/01/2001 14:34:10 nsa-gw.cisco.com 1
03/01/2001 14:39:10 nsa-gw.cisco.com 1
03/01/2001 14:44:10 nsa-gw.cisco.com 1
03/01/2001 14:49:10 nsa-gw.cisco.com 1
03/01/2001 14:54:10 nsa-gw.cisco.com 1
03/01/2001 14:59:10 nsa-gw.cisco.com 1
03/……
```

Как только выходные данные тестового файла появятся на вашей станции UNIX, вы сможете перенести их на свой компьютер, используя протокол передачи файлов FTP.

Сбор данных может осуществляться и при помощи собственных сценариев. Чтобы сделать это, выполните **snmpget** для OID ЦП каждые пять минут и формируйте дамп результатов в.csv file.

Шаг 4. : Проанализируйте данные для определения порогов

Теперь, когда у вас есть некоторые данные, можно начать анализировать их. На данном этапе базового анализа определяются параметры порога, позволяющие точно измерить производительность или надежность при отсутствии избыточных предупреждений при мониторинге пороговых значений. Проще всего будет импортировать данные в электронную таблицу, например Microsoft Excel, и начертить точечную диаграмму. Этот метод делает очень легким видеть, сколько раз конкретное устройство создало бы предупреждение исключения при мониторинге его для определенного порога. Не желательно включить пороги, не делая срока, так как это может создать аварийные штормы от устройств, которые превысили порог, который вы выбрали.

Для импорта тестового файла в Электронную таблицу Excel откройте Excel и выберите **File> Open** и выберите файл данных.

Затем приложение Excel предлагает выбрать файл для импорта.

По завершении процедуры импортированный файл должен быть аналогичен следующему примеру.

Диаграмма рассеяния позволяет вам более легко визуализировать, как различные параметры порога работали бы на сеть.

Чтобы создать график разброса, выделите столбец C в импортированном файле и щелкните значок мастера диаграмм. Тогда следуйте указаниям мастера создания диаграмм, чтобы построить график разброса.

В Шагах мастера создания диаграмм 1, как показано ниже, выбирают вкладку **Standard Types** и выбирают **XY (Рассеяние)** тип диаграммы. **Нажмите кнопку Next.**

В Шагах мастера создания диаграмм 2, как показано ниже, выбирают вкладку **Data Range** и

выбирают диапазон данных и **Параметр столбцов**. **Нажмите кнопку Next.**

В Шагах мастера создания диаграмм 3, как показано ниже, вводят заголовок диаграммы и значения оси X и y, и затем нажимают **Next**.

В Шагах мастера создания диаграмм 4, выберите, хотите ли вы диаграмму рассеяния на новой странице или как объект на существующей странице.

Нажмите **Finish** для размещения диаграммы в желаемое местоположение.

Анализ возможностей?" Анализ

Теперь можно использовать точечную диаграмму для анализа. Однако перед переходом, необходимо задать следующие вопросы:

- Что рекомендует поставщик (в этом примере поставщик – это Cisco) в качестве порога для этой переменной MIB?В целом Cisco рекомендует, чтобы центральный маршрутизатор не превышал 60-процентную среднюю загрузку ЦПУ. Шестьдесят процентов были выбраны, потому что маршрутизатору нужны некоторые издержки в случае, если это испытывает проблему, или сеть имеет некоторые сбои. Cisco оценивает, что центральному маршрутизатору нужны приблизительно 40 издержек процента cpu в случае, если протокол маршрутизации должен повторно вычислить или повторно сойтись. Эти проценты изменяются в зависимости от используемых протоколов, топологии и стабильности сети.
- Что будет, если пороговое значение установить на уровне 60 процентов?Если вы будете чертить линию через диаграмму рассеяния горизонтально в 60, то вы будете видеть, что ни одна из точек данных не превышает 60 процентов использования CPU. Таким образом, порог 60 наборов на ваших станциях системы управления сетью (NMS) не будет выделять сигнал порога во время периода опроса. Процент от 60 приемлем для этого маршрутизатора. Однако заметьте в диаграмме рассеяния, что некоторые точки данных близко к 60. Было бы хорошо знать, когда маршрутизатор приближается к 60-процентному порогу, таким образом, можно знать заранее это, ЦП приближается к 60 процентам, и имейте план относительно того, что сделать, когда это достигает той точки.
- Что, если я установил порог к 50 процентам?Считается, что этот маршрутизатор достиг 50 процентов загрузки четыре раза во время этого цикла опроса и будет генерировать сигнал порога каждый раз. Этот процесс становится более важным при рассмотрении *групп маршрутизаторов* для наблюдения то, что сделали бы другие параметры порога. Например, "Что, если я установил порог в 50 процентах для всей базовой сети? Как видно, тяжело выбрать только одно число.

Пороговое значение ЦПУ, "что, если" анализ

Одна стратегия, которую можно использовать для создания этого легче, является Готовым, Набором, Пойдите методология порогов. В этой методике последовательно используется три пороговых значения.

- Готово вЪ" – порог, задаваемый в качестве прогнозирующего фактора будущих операций с устройствами для обеспечения их нормальной работы

- Установите порог, который будет использоваться в качестве раннего индикатора и напоминать о начале ремонта, изменения конфигурации или обновления
- Пойдите — порог, которому вы и/или поставщик верите, условие отказа и требует, чтобы некоторое действие восстановило его; в данном примере это - 60 процентов

В следующей таблице показана стратегия Ready, Set, Go.

Threshold	Действие	Результат
45 процентов	Займитесь расследованиями далее	Список опций для планов действий
50 процентов	Сформулируйте план действий	Список шагов в план действий
60 процентов	План действий внедрения	Маршрутизатор больше не превышает пороговое значение. Назад к Режиму готовности

Методология "Ready, Set, Go" изменяется в исходных схемах базового анализа, описанных ранее. На следующей диаграмме показана измененная схема базового анализа. Если можно определить другие точки пересечения на диаграмме, у вас теперь есть больше времени, чтобы запланировать и реагировать, чем вы сделали прежде.

Заметьте, что в этом процессе, внимание сосредоточено на исключениях в сети и не касается других устройств. Предполагается, что, пока устройства ниже порогов, они в порядке.

Если у вас будет эта мысль шагов из начала, то вы будете хорошо подготовлены к хранению здоровой сети. Выполнение этого типа планирования также чрезвычайно полезно для планирования бюджета. Если вы знаете то, что ваши лучшие пять **идут** маршрутизаторы, ваши средние маршрутизаторы **набора**, и ваши нижние **готовые маршрутизаторы**, можно легко запланировать то, в каком количестве бюджета вы будете нуждаться для обновлений на основе того, какие маршрутизаторы они и каковы ваши опции плана действий. Та же стратегия может использоваться для ссылок глобальной сети (WAN) или любого другого OID MIB.

Шаг 5. : Исправьте определенные неотложные проблемы

Это один из самых простых этапов процесса базового анализа. **Как только определены устройства, выходящие за пределы порогового значения, необходимо составить план действий, чтобы устройства не превышали порог.**

Можно открыть случай с Центром технической поддержки (TAC) Cisco или связаться Системным инженером для доступных параметров. Вы не должны предполагать, что возвращение вещей под порогом будет стоить вам денег. Многие проблемы с CPU можно разрешить с помощью изменений конфигурации, чтобы убедиться, что все процессы работают с максимальной эффективностью. Например, некоторые Списки контроля доступа (ACL) могут сделать процессор маршрутизатора выполненным очень причина высокой загрузки к пути, который пакеты берут через маршрутизатор. В некоторых случаях можно внедрить Коммутацию сетевых потоков, чтобы изменить путь перенаправления пакетов и

уменьшить влияние ACL на ЦП. Вне зависимости от задач, на данном этапе необходимо вернуть все маршрутизаторы в рамки пороговых значений, чтобы установить пороговые значения позже, не рискуя переполнить станции NMS чрезмерным количеством сигналов порога.

Шаг 6: Тестирование функций проверки пороговых значений

На данном этапе предусматривается проверка пороговых значений в лабораторных условиях с помощью средств, которые будут использованы рабочей сети. Есть два способа мониторинга порогов. Необходимо выбрать наиболее подходящий для Вашей сети способ.

- Опросите и сравните метод с помощью платформы SNMP или другого средства мониторинга SNMPЭтот метод использует больше пропускной способности сети для трафика опроса и приводит циклы обработки в рабочее состояние на вашей платформе SNMP.
- Используйте конфигурации сигналов тревоги и событий удаленного мониторинга (RMON) на маршрутизаторах, чтобы они отправляли сигнал тревоги только при превышении порогаЭтот метод снижает использование пропускной способности сети, но и увеличивает использование памяти и CPU на маршрутизаторах.

Реализация Порога с помощью SNMP

Для устанавливания способа SNMP с помощью NNM HP OpenView выберите **Options> Data Collection и Thresholds**, как вы сделали, когда вы устанавливаете первоначальный последовательный опрос. **В этот раз в меню набора выберите Хранить, Проверить пороговые величины, а не Хранить, Нет пороговых величин**. После того, как вы установите порог, можно повысить загрузку ЦПУ на маршрутизаторе путем передачи ему множественных эхо-запросов и/или множественных обходов SNMP. Может потребоваться понижение порога, если невозможно разогнать CPU так, чтобы преодолеть порог. В любом случае необходимо гарантировать, что работает пороговый механизм.

Одним из ограничений использования этого метода является тот факт, что нельзя одновременно внедрить несколько порогов. Понадобится три SNMP платформы, чтобы установить три разных синхронных порога. Программные средства, такие как Работоспособность сети Concord и Trinagy TREND позволяют несколька порогов для того же экземпляра OID.

Если ваша система может только обработать один порог за один раз, можно полагать, что Готовые, Набор, Идут стратегия последовательной формой. **Если порог готовности достигается постоянно, выполните анализ и увеличьте пороговое значение до установленного для этого устройства уровня. Когда установленный уровень достигается непрерывно, создайте план действий и увеличьте пороговое значение до уровня запуска этого устройства**. Затем, когда предел GO уже несколько раз достигнут, приступайте к своему плану действий. Это прием должен сработать так же, как и метод трех одновременных порогов. Просто требуется немного больше времени, изменяя параметры порога платформы SNMP.

Реализация Порога с помощью Сигнального оповещения "rmon" и События

Используя конфигурации предупреждений и событий RMON, можно настроить

маршрутизатор для контроля собственных нескольких порогов. Если маршрутизатор обнаруживает условие превышения порога, он отправляет ловушку SNMP платформе SNMP. В конфигурации маршрутизатора должен быть настроен приемник прерываний SNMP, чтобы прерывание могло быть перенаправлено. Имеет место корреляция между аварийным сигналом и событием. Сигнал тревоги проверяет OID для данного порога. Если порог достигнут, процесс обработки аварийных сигналов запускает event process, который может или передать сообщение прерывания SNMP, создать запись журнала RMON или обоих. Для большего количества подробности об этой команде посмотрите Сигнальное оповещение "rmon" и Команды конфигурации событий.

Следующие команды конфигурации маршрутизатора выдают на мониторе маршрутизатора cpmCPUTotal5min каждые 300 секунд. Это запустит событие 1, если ЦП превысит 60 процентов и запустит событие 2, когда ЦП переключается на 40 процентов. В обоих случаях сообщение прерывания SNMP будет передаваться Станции NMS с сообществом частную строку.

Для быстрой настройки используйте все следующие операторы конфигурации.

```
rmon event 1 trap private description "cpu hit60%" owner jharp
rmon event 2 trap private description "cpu recovered" owner jharp
rmon alarm 10 cpmCPUTotalTable.1.5.1 300 absolute rising 60 1 falling 40 2 owner jharp


rmon event 3 trap private description "cpu hit50%" owner jharp
rmon event 4 trap private description "cpu recovered" owner jharp
rmon alarm 20 cpmCPUTotalTable.1.5.1 300 absolute rising 50 3 falling 40 4 owner jharp

rmon event 5 trap private description "cpu hit 45%" owner jharp
rmon event 6 trap private description "cpu recovered" owner jharp
rmon alarm 30 cpmCPUTotalTable.1.5.1 300 absolute rising 45 5 falling 40 6 owner jharp
```

Приведенный ниже пример содержит результаты выполнения команды show rmon alarm, настроенной с помощью описанных выше операторов.

```
zack#sh rmon alarm Alarm 10 is active, owned by jharp Monitors cpmCPUTotalTable.1.5.1 every 300
second(s) Taking absolute samples, last value was 0 Rising threshold is 60, assigned to event 1
Falling threshold is 40, assigned to event 2 On startup enable rising or falling alarm Alarm 20
is active, owned by jharp Monitors cpmCPUTotalTable.1.5.1 every 300 second(s) Taking absolute
samples, last value was 0 Rising threshold is 50, assigned to event 3 Falling threshold is 40,
assigned to event 4 On startup enable rising or falling alarm Alarm 30 is active, owned by jharp
Monitors cpmCPUTotalTable.1.5.1 every 300 second(s) Taking absolute samples, last value was 0
Rising threshold is 45, assigned to event 5 Falling threshold is 40, assigned to event 6 On
startup enable rising or falling alarm
```

В следующем примере показы выходные данные выполнения команды show rmon event.

```
zack#sh rmon event Event 1 is active, owned by jharp Description is cpu hit60% Event firing
causes trap to community private, last fired 00:00:00 Event 2 is active, owned by jharp
Description is cpu recovered Event firing causes trap to community private, last fired 02:40:29
Event 3 is active, owned by jharp Description is cpu hit50% Event firing causes trap to
community private, last fired 00:00:00 Event 4 is active, owned by jharp Description is cpu
recovered Event firing causes trap to community private, last fired 00:00:00 Event 5 is active,
owned by jharp Description is cpu hit 45% Event firing causes trap to community private, last
fired 00:00:00 Event 6 is active, owned by jharp Description is cpu recovered Event firing
causes trap to community private, last fired 02:45:47
```

Возможно, потребуется испытать оба этих метода для выбора того из них, которых лучше работает в конкретном окружении. Возможно, что отлично будет работать комбинация этих методов. В любом случае тестирование должно быть сделано в лабораторной среде, чтобы гарантировать, что все работает правильно. После тестирования в лабораторной работе ограниченное развертывание на небольшой группе маршрутизаторов позволит вам

тестировать процесс передачи предупреждений к вашему Центру Операций.

В этом случае необходимо будет понизить пороги для тестирования процесса: Попытка искусственно повысить ЦП на производственном маршрутизаторе не рекомендуется. Следует также убедиться, что во время получения предупреждений станциями NMS в центре управления существует политика эскалации для информирования о превышении порога устройствами. Эти конфигурации проверены в лабораторной среде Cisco IOS версии 12.1(7). При обнаружении с какими-либо проблемами необходимо свериться с Разработкой Cisco или Системными инженерами, чтобы видеть, есть ли у вас дефект в вашей версии IOS.

Шаг 7: Контроль порога внедрения с помощью SNMP или RMON

Если в лаборатории имеется тщательно испытанный пороговый контроль при ограниченном развертывании, то можно применить пороги по всей базовой сети. Впоследствии можно систематически повторять данный процесс базового анализа для других важных переменных MIB в сети, таких как буферы, свободная память, ошибки CRC, потеря ячеек ATM и т.д.

При использовании Сигнального оповещения "rmon" и конфигураций событий можно теперь прекратить опрашивать от Станции NMS. Это позволит снизить нагрузку на сервер NMS и уменьшить объем данных последовательного опроса в сети. Путем систематического прохождения через этого процесса для индикаторов исправности важной сети вы могли легко перейти к сути дела, который сетевое оборудование контролирует само с помощью Сигнального оповещения "rmon" и События.

Дополнительные базы управляющей информации (MIB)

После обучения этого процесса можно хотеть исследовать другие MIB к сроку и монитору. Следующие подразделы содержат краткий список OID и их описания.

Базы управляющей информации (MIB) маршрутизатора

Характеристики памяти очень полезны в определении состояния маршрутизатора. Исправный маршрутизатор должен почти всегда иметь доступное пространство буфера, с которым можно работать. Если маршрутизатор начинает исчерпывать пространство буфера, ЦП должен будет работать тяжелее, чтобы создать новые буферы и попытаться найти буферы для поступления и исходящих пакетов. Подробное обсуждение буферов выходит за рамки этого документа. Однако как правило исправный маршрутизатор должен иметь очень немногих, если таковые имеются, буферные неудачи и не должен иметь никаких ошибок буфера или состояния памяти, не содержащей нулей.

Объект	Описание	OID
ciscoMemoryPoolFree	Число байтов из пула памяти, не используемых в текущий момент на управляющем устройстве	1.3.6.1.4.1.9.9.48.1.1.1.6
ciscoMemoryP	Наибольшее число	1.3.6.1.4.1.9

oolLargestFree	непрерывных байтов от пула памяти, которые являются не используемыми в настоящее время	.9.48.1.1.1.7
bufferElMiss	Число недостающих элементов буфера	1.3.6.1.4.1.9.2.1.12
bufferFail	Количество отказов размещения буферов	1.3.6.1.4.1.9.2.1.46
bufferNoMem	Номер буфера, вызвавшего сбой из-за отсутствия памяти	1.3.6.1.4.1.9.2.1.47

Базы управляющей информации на коммутаторах Catalyst

Объект	Описание	OID
cpmCPUTotal5min	Полный Процент занятости ЦПУ в последнем пятиминутном периоде. Этот объект исключает объект avgBusy5 из OLD-CISCO-SYSTEM-MIB	1.3.6.1.4.1.9.9.109.1.1.1.5
cpmCPUTotal5sec	Полный Процент занятости ЦПУ в последнем пятисекундном периоде. Данный объект делает объект busyPer в OLD-CISCO-SYSTEM-MIB устаревшим	1.3.6.1.4.1.9.9.109.1.1.1.3
sysTraffic	Процент загруженности полосы пропускания для предыдущего интервала опроса	1.3.6.1.4.1.9.5.1.1.8
sysTrafficPeak	Пиковое значение измерителя трафика со времени последнего обнуления счетчиков портов или запуска системы	1.3.6.1.4.1.9.5.1.1.19
sysTrafficPeaktime	Время (в сотых секунды), прошедшее с тех пор, как было достигнуто значение пикового трафика по счетчику	1.3.6.1.4.1.9.5.1.1.20
portTopNUtilization	Использование порта в системе	1.3.6.1.4.1.9.5.1.20.2.1.4
portTopNBufferOverFlow	Число переполнений буфера порта в системе	1.3.6.1.4.1.9.5.1.20.2.1.10

Базы управляющей информации (MIB) последовательных соединений

Объект	Описание	OID
locIfInputQueueDrops	Число отброшенных пакетов из-за переполнения очереди входа	1.3.6.1.4.1.9.2.2.1.1.26
locIfOutputQueueDrops	Число отброшенных пакетов из-за переполнения очереди вывода	1.3.6.1.4.1.9.2.2.1.1.27
locIfInCRC	Количество входящих пакетов, содержавших ошибки циклической контрольной суммы	1.3.6.1.4.1.9.2.2.1.1.12

Команды "RMON Alarm" и "Event Configuration"

Сигналы тревоги

Сигналы тревоги RMON можно настроить со следующим синтаксисом:

rmon alarm *number variable interval* **{delta | absolute} rising-threshold** *value* **[***event-number***] falling-threshold** *value* **[***event-number***] [owner** *string***]**

Element	Описание
номер	Номер аварийного сигнала, идентичный индексу alarmIndex в таблице alarmTable в удаленном мониторинге MIB.
переменная	Объект MIB для наблюдения, транслирующий в alarmVariable, использованную в alarmTable RMON MIB.
интервал	Время (в секундах), в течение которого сигнал проверяет переменную MIB; его значение идентично alarmInterval, используемому в alarmTable RMON MIB.
дельта	Тестирует изменение между переменными MIB, которое влияет на alarmSampleType в alarmTable RMON MIB.
абсолютный	Проверяет каждую переменную MIB напрямую, что влияет на alarmSampleType в alarmTable RMON MIB.
значение верхнего порога	Значение, в котором инициирован сигнал тревоги.

event-number	(Необязательно) число событий для инициирования, когда повышение или нижний порог превышают свой предел. Данное значение идентично alarmRisingEventIndex или alarmFallingEventIndex в таблице alarmTable базы RMON MIB.
значение нижнего порога	Значение, при котором сбрасывается сигнал.
строка owner	Определяет владельца этого события, который совпадает с alarmOwner в таблице alarmTable базы RMON MIB (необязательно).

События

События RMON могут быть настроены со следующим синтаксисом:

rmon event *number* **[log] [trap** *community***] [description** *string***] [owner** *string***]**

Element	Описание
номер	Назначенное число событий, которое идентично eventIndex в eventTable в RMON MIB.
журнал	(необязательный) Создает запись в журнале RMON, когда событие запускается, и устанавливает тип события в MIB RMON, чтобы регистрировать или регистрировать-и-перехватывать.
сообщество прерываний	строка имени и пароля SNMP, использованный для данной ловушки (необязательно). Настраивает значение eventType в RMON MIB для этой строки или как snmptrap или как журнал-и-trap-сообщение. Это значение идентично eventCommunityValue в eventTable в RMON MIB.
description string	Указывает описание события, которое совпадает с описанием события в таблице eventTable базы RMON MIB (необязательно).
строка owner	(Необязательно) Владелец этого события, который совпадает с eventOwner в таблице eventTable базы RMON MIB.

Реализация аварийных сигналов и событий RMON

Для получения дальнейшей информации о Применении оповещений и событий RMON, считайте Раздел применения оповещений и событий RMON Описания технологических

решений *Оптимальных методов Систем управления сетью*.

Дополнительные сведения

- Техническая поддержка и документация – Cisco Systems