Управление сетями с высоким уровнем доступности Cisco IOS: Рекомендации и Описание технологических решений

Содержание

Введение

Развертывание и поддержание надежного программного обеспечения Cisco IOS являются приоритетом в сегодняшней критически важной для бизнеса сетевой среде, которая требует, чтобы возобновленная Cisco и фокус на заказчика достигли безостановочной доступности. В то время как компания Cisco основной акцент делает на высокое качество программного обеспечения, специалисты по организации и сопровождению сетей должны сосредоточиться на наилучшем управлении возможностями программного обеспечения Cisco IOS. Целью является более высокая доступность и эффективность управления программным обеспечением. Этот метод представляет собой комбинированное партнерство для совместного использования, изучения и внедрения лучших методов управления программным обеспечением.

Этот документ предоставляет эффективную рабочую структуру методов управления Cisco IOS и для Предприятия и для Клиентов поставщика услуг, что справка способствует надежности улучшенного ПО, уменьшенной сложности сети и повышению сетевой доступности. Эта структура также помогает улучшить эффективность управления программами путем определения областей ответственности, а также перекрытий в проверках управления ПО и проверках достоверности между официальными возможностями ПО Cisco и клиентской основой Cisco.

Обзор рекомендаций по Cisco IOS

Следующие таблицы содержат обзор оптимальных методов использования Cisco IOS. Эти таблицы можно использовать в качестве обзора лучших методов управления, контрольного листа сравнительного анализа текущих приемов управления Cisco IOS или основы для создания процессов, связанных с управлением Cisco IOS.

Таблицы определяют четыре компонента жизненного цикла управления Cisco IOS. Каждая таблица запускается со стратегии и сводки программных средств для определенной области жизненного цикла. После стратегии и программных средств сводка является определенными оптимальными методами, которые применяются только к определенной области жизненного цикла.

При планировании - Построение Платформы управления Cisco IOS — Планирование является начальной фазой управления Cisco IOS, должен был помочь организации определять, когда к обновлению ПО, где обновить, и какой процесс будет использоваться, чтобы протестировать и проверить возможные образы.

Практические рекомендации	Подробность
Стратегия и инструменты планирования Cisco IOS	Начало работы с планированием управления Cisco IOS начинается с честной оценки существующей практики, постановки реальных задач и планирования проекта.
Определения циклов	Определяет непротиворечивость программного обеспечения where, может быть поддержан. Модель ПО может быть

версий программного обеспечения	определена как группировка уникальных версий ПО, дифференцируемая от других областей уникальной географией, платформами, модулем или требованиями к характеристикам.
Цикл обновления и соответствующие определения	Определения цикла обновления – это набор основных действий по обеспечению качества и управлению изменениями, используемых для определения момента инициации цикла обновления программного обеспечения.
Процесс получения сертификата	Шаги процесса получения сертификата должны включать идентификацию дорожки, определения цикла обновления, управление кандидатами, тестирование/проверку и по крайней мере некоторое использование пилотного производства.

Дизайн - Выбор и Проверка Версий IOS — Наличие четко определенного процесса для выбора и проверки версий Cisco IOS помогают организации уменьшать простой простой из-за неуспешных попыток модернизации и незапланированных ошибок ПО.

Практические рекомендации	Подробность
Стратегия и инструменты выбора и проверки Cisco IOS	Определите процессы для выбора, тестирования и проверки новых ПО Cisco IOS версий. Сюда относится сетевая лаборатория тестирования, эмулирующая производственную сеть
Управление кандидатами	Управление кандидатами является идентификацией требований к версии программного обеспечения и потенциальных рисков для конкретного оборудования и включило наборы функций.
Тестирование и проверка	Тестирование и проверка – это критически важный аспект управления программным обеспечением и бесперебойной работы сети. Надлежащее лабораторное испытание может значительно уменьшить время простоя производства, помочь к персоналу технической поддержки сети железных дорог и помочь в оптимизации процессов

	реализации сети.

Реализация - Быстрые и Успешные Развертывания Cisco IOS — Четко определенные процессы внедрения разрешают организацию быстро, и успешно разверните новые ПО Cisco IOS версии.

Практические рекомендации	Подробность
Стратегия и инструменты развертывания Cisco IOS	Основная стратегия развертывания Cisco IOS заключается в выполнении финальной сертификации через опытный процесс и быстрое развертывание с использованием средств обновления и проверенных процессов реализации.
Пилотный процесс	Для уменьшения потенциальной незащищенности и более безопасно перехватывать любые остающиеся проблемы производства, пробный релиз программы рекомендуется. Индивидуальный пилотный план должен включать выбор пилотного плана, его длительность и показатели.
Реализация	После завершения пилотной фазы должна начаться фаза реализации Cisco IOS. Этап внедрения предусматривает несколько действий по обеспечению успешного обновления программного обеспечения и эффективности его работы, включая медленную загрузку, итоговую сертификацию, подготовку к обновлению, автоматизацию обновления и итоговую проверку.

Операции - Управление Реализацией Cisco IOS Высокой доступности — Оптимальные методы для операций Cisco IOS включает управление версиями программного обеспечения, Управление системным журналом Cisco IOS, управление проблемами, стандартизацию конфигурации и управление доступностью.

Практические рекомендации	Подробность
Стратегии и инструменты управле	Первая стратегия операций Cisco IOS состоит в том, чтобы поддержать среду максимально простой, избежав изменения в конфигурации и версиях Cisco IOS. Вторая стратегия является способностью

ния Cisco IOS	определить и быстро решить сбои сети.
Контроль версий программного обеспечения	Контроль версий программного обеспечения - это процесс внедрения только стандартных версий программного обеспечения с контролем сети для выявления и возможного изменения программного обеспечения в связи с несовместимостью версий.
Упреждающее управление системным журналом	Коллекции системного журнала, контроля и анализа являются процессами управления ошибками, которые рекомендуется использовать для разрешения специфических для сети Cisco IOS проблем, которые сложно или невозможно обнаружить другими способами.
Решение проблем	Подробные процессы управления проблемами, которые определяют распознавание ошибки, сбор сведений и хорошо проанализированный путь решения. Эти данные могут использоваться для определения корневой причины.
Стандартизация конфигурации	Стандарты конфигурации представляют практику создания и поддержания стандартных параметров глобальной конфигурации через подобные устройства и сервисы, приводящие к предприятию широкая согласованность глобальной конфигурации.
Управление доступностью	Управление доступностью является процессом повышения качества с помощью доступности сети в качестве метрики повышения качества.

Обзор процесса управления жизненным циклом программного обеспечения

Управление жизненным циклом Программного обеспечения Cisco IOS определено как набор планирования, дизайна, реализации и в рабочем состоянии процессов, которые рекомендуются для реализаций надежного программного обеспечения и сети с высоким уровнем доступности. Это включает в себя процессы выбора, проверки и использования версий Cisco IOS в сети.

Цель управления жизненным циклом программного обеспечения Cisco IOS состоит в том, чтобы улучшиться, доступность сети путем понижения возможности производства определила ошибки ПО, или программное обеспечение отнеслось изменение/сбои при обновлении. Лучшие способы решения проблем, описанные в этом документе, доказали

возможность уменьшить подобные дефекты и исправить ошибки на основе практического опыта многих клиентов Cisco и группы Cisco Advanced Services. Управление жизненным циклом оборудования может поначалу увеличить расходы, однако общая стоимость владения может быть уменьшена благодаря снижению числа сбоев и более усовершенствованным механизмам развертывания и поддержки.

Планирование и построение структуры управления Cisco IOS

Планирование – начальная фаза управления Cisco IOS, во время которой организация получает помощь в определении времени обновления программного обеспечения, обновляемых компонентов и процесса, который будет использоваться для тестирования и проверки потенциальных образов.

Оптимальные методы включают определения пути версии программного обеспечения, цикл обновления и определения и создание процесса получения сертификата внутреннего программного обеспечения.

Стратегия и инструменты планирования Cisco IOS

Начните управление Cisco IOS, планирующее с честной оценки существующей практики, постановки реальных задач и планирования проекта. Самооценка может быть проведена путем сравнения лучших методов, описанных в данном документе, с процессами в вашей организации. Основные вопросы должны включать придерживающееся:

- Моя организация имеет процесс получения сертификата программного обеспечения этим, которое включает программное обеспечение, testing/validation?
- Моя организация имеет стандарты программного обеспечения Cisco IOS с ограниченным количеством версий Cisco IOS, работающих в сети?
- Моя организация испытывает затруднения при определении, когда обновить программное обеспечение Cisco IOS?
- Моя организация испытывает затруднения при развертывании нового ПО Cisco IOS программного обеспечения оба эффективно и продуктивно?
- Моя организация имеет проблемы со стабильностью Cisco IOS после развертываний, которые серьезно влияют на стоимость простоя?

После оценки ваша организация должна начать определять цели для управления программного обеспечения Cisco IOS. Начните с формирования комплексной команды менеджеров и/или руководителей из групп архитектурного планирования, инженерной группы, группы внедрения и эксплуатации, чтобы они помогли определить цели Cisco IOS, а также разработать проект усовершенствования процесса. Цель первых встреч – определить общие задачи, роли и обязанности, назначить объекты действий и установить начальные сроки проекта. Кроме того, определите критичные факторы успеха и метрики для определения преимуществ управления программным обеспечением. Возможные метрики включают:

- доступность (из-за проблем программного обеспечения)
- стоимость обновлений программного обеспечения
- время, необходимое для обновлений
- количество версий программного обеспечения, использующихся в производстве
- успех/интенсивность отказов изменения обновления ПО

В дополнение к полному планированию платформы управления Cisco IOS некоторые организации также определяют продолжающиеся плановые совещания программного обеспечения для появления ежемесячно или ежеквартально. Цель этих совещаний состоит в том, чтобы рассмотреть текущее развертывание ПО и начать планировать любые новые требования к программному обеспечению. В планирование может также входить перепроверка или изменение текущих процессов программного управления, или простое распределение ролей и ответственности между различными фазами программного управления.

В инструменты на этапе планирования входят исключительно инструменты управления инвентаризации программного обеспечения. Менеджер по инвентарю Resource Manager Essentials (RME) CiscoWorks 2000 является основным программным средством, используемым в этой области. CiscoWorks2000 RME Inventory Manager упрощает управление версиями маршрутизаторов Cisco и коммутаторов через находящиеся на web средства создания отчетов, которые сообщают и устройства Cisco IOS вида на основе версии программного обеспечения, платформы устройства, размера памяти и имени устройства.

Определения циклов версий программного обеспечения

Первый оптимальный метод планирования управления программного обеспечения Cisco IOS определяет непротиворечивость программного обеспечения where, может быть поддержан. Модель ПО определена как группировка уникальных версий ПО, дифференцируемая от других областей уникальной географией, платформами, модулем или требованиями к характеристикам. При оптимальном варианте сеть должна работать с единой версией программы. Это значительно понижается, управление программным обеспечением отнеслось затраты и предоставляет последовательную и легко управляемую среду. Однако действительность - то, что большинство организаций должно выполнить несколько версий в сети из-за функции, платформы, миграции и проблем доступности в определенных областях. Во многих случаях та же версия не работает на разнотипные платформы. В других случаях организация не может ждать одной версии для поддержки всех их требований. Целью является идентификация наименьшего количества слежений программного обеспечения для сети с условием тестирования/подтверждения, сертификации и соответствия требованиям обновления. Во многих случаях организация может иметь немного больше дорожек для понижения тестирования/проверки, сертификации, и обновление стоит в целом.

Первое различие — это поддержка платформы. Обычно для коммутаторов сетей LAN, WAN, а также центральных и граничных маршрутизаторов используются раздельные программные средства отслеживания. Для конкретных функций или служб могут потребоваться другие средства отслеживания программного обеспечения, такие как переключение цифрового канала (DLSw), качество обслуживания (QoS) или IP-телефония, особенно если эти требования могут быть локализованы внутри сети.

Другой критерии является надежностью. Много организаций пытаются выполнить большую часть надежного программного обеспечения к ядру сети и ЦОД, при предложении более новых дополнительных характеристик или аппаратной поддержки, к краю. С другой стороны, масштабируемость или функции пропускной способности часто больше всего необходимы в средах ЦОД или ядре. Другие модели могут потребоваться для определенных платформ, таких как более крупные узлы распределения, имеющие другую платформу маршрутизатора WAN. Следующая таблица представляет собой пример определения отслеживания программного обеспечения для большой корпоративной организации.

Дорожка	Область	Аппаратные платформы	Функции	Версия Cisco IOS	Статус сертификации
1	Коммутация ядра LAN	6500	QoS	12.1E (A8)	Тестирование
2	Коммутатор доступа к локальной сети	2924XL 2948XL	Протокол обнаружения однонаправленной связи (UDLD), Spanning Tree Protocol (STP)	12.0 (5.2) XU	Сертифицируемый 01.03.01
3	Распределение LAN / доступ	5500 6509	Супервизор 3	5.4 (4)	Сертифицируемый 01.07.01
4	Модульный коммутатор с функциями маршрутизатора (RSM) коммутатора распределения	RSM	Маршрутизация первоочередного открытия кратчайших маршрутов (OSPF)	12.0 (11)	Сертифицируемый 04.03.02
5	Распределение головного узла WAN	7505 7507 7204 7206	Frame Relay OSPF	12.0 (11)	Сертифицируемый 01.11.01
6	Доступ через WAN	2600	Frame Relay OSPF	12.1 (8)	Сертифицируемый 01.06.01
7	Возможность соединения с IBM	3600	Головной узел Протокола SDLC	11.3 (8) T1	Сертифицируемый 11/1/00

Распределение дорожек также может меняться с течением времени. Во многих случаях

функции или аппаратная поддержка могут интегрироваться в большее количество версий программного обеспечения магистрали, позволяющих другие дорожки в конечном счете мигрировать вместе. После установки определений отслеживания могут использоваться другие определенные процессы для обеспечения согласованности и корректности новых версий. Кроме того, постоянно требуются определения отслеживаний. В любое время новая характеристика, сервис, аппаратные средства или требование к модулю определен, новый трек нужно рассмотреть.

Организации, желающие инициировать процесс дорожки, должны запуститься с недавно определенных требований дорожки, или в некоторых случаях, проекты стабилизации для существующих сетей. Организация может также иметь некоторую идентифицируемую общность с существующими версиями программного обеспечения, которые могут сделать текущее определение пути возможным. В большинстве случаев, если у клиента есть достаточная устойчивость сети, быстрый перенос к определенным версиям не требуется. Сетевая архитектура или инженерная группа, обычно владеет процессом определения пути. В некоторых случаях одно частное лицо может быть ответственно за определения пути. В других случаях руководители проекта ответственны за разработку требований к программному обеспечению и определений нового трека на основе отдельных проектов. Это - также хорошая идея рассмотреть определения пути на ежеквартальной основе, чтобы определить, требуются ли новые треки, или если старые дорожки требуют консолидации или обновления.

Как показывает практика, организации, которые используют средства управления версиями ПО, успешно сокращают количество версий ПО в рабочей сети. Обычно это улучшает устойчивость программного обеспечения и надежность всей сети.

Цикл обновления и соответствующие определения

Определения циклов обновления представляют собой описания основных действий в области управления программным обеспечением и внесением изменений, которые позволяют установить, когда следует начинать очередной цикл модернизации программного обеспечения. Определения цикла обновления позволяют организации должным образом планировать цикл обновления программного обеспечения и выделять необходимые ресурсы. Без определения цикла обновлений организация обычно испытывает нарастание проблем с надежностью программного обеспечения в связи с функциональными требованиями в последних устойчивых версиях. Другое воздействие могло быть организацией, пропускающей возможность должным образом протестировать и проверить новую версию, прежде чем будет требоваться производственное использование.

Важный аспект этой практики определяет, когда и до какой степени должны инициироваться процессы планирования программного обеспечения. Это - то, вследствие того, что главная причина неполадок программного обеспечения включает функцию, сервис или быстродействие оборудования в производстве без должного внимания, или обновляет к новой ПО Cisco IOS версии без обсуждений управления программным обеспечением. Другая проблема не обновляет. Путем игнорирования обычных циклов программного обеспечения и требований, много клиентов сталкиваются со сложной задачей обновления программного обеспечения через многие другие основные релизы. Трудности могут быть связаны с размером образа, изменениями стандартного поведения, изменениями интерпретатора командного уровня (CLI) и изменениями протокола.

Cisco рекомендует четко определенный цикл обновления, на основе оптимальных методов, как определено в этой газете, чтобы инициироваться каждый раз, когда требуются новая

основная характеристика, сервис или аппаратная поддержка. Следует проанализировать (на основе риска) степень сертификации и тестирования/проверки достоверности, чтобы определить точные требования к тестированию/проверке достоверности. Анализ степени риска можно провести на основе географического расположения, логического расположения (уровень доступа, уровень распределения или магистральный уровень) или ожидаемого количества затрагиваемых пользователей. Если основная характеристика или быстродействие оборудования содержатся в текущем релизе, некоторые хороша организованные процесс цикла обновления должны также инициироваться. Если функция относительно незначительна, рассмотрите риск и затем решите, какие процессы должны инициироваться. Кроме того, программное обеспечение должно быть обновлено через два года или меньше помочь гарантировать, что ваша организация остается относительно текущей и что процесс обновления не является слишком громоздким.

Клиенты должны также рассмотреть факт, что никакие исправления ошибки не будут сделаны к версиям ПО, которые передали статус окончания срока эксплуатации (EOL). Следует также обратить внимание на бизнес-требования, поскольку многие среды способны выдерживать и даже приветствуют более доработанные версии с краткими процессами тестирования/проверки или вообще без них и небольшим итоговым временем простоя. Клиенты должны также считать более новые данные собранными в операциях Релиза Cisco при рассмотрении их тестовых требований. Анализ дефектов и основных причин показал, что большая часть основных причин дефекта была результатом разработчиков, кодирующих в затронутой области ПО. Это означает, что, если организация добавляет определенную характеристику или модуль к их сети в существующем выпуске, существует вероятность испытания дефекта, отнесенного к той функции или модулю, но намного более низкая вероятность, что новая характеристика, аппаратные средства или модуль повлияют на другие области. Эти данные должны позволять организациям снизить требования к проверке при добавлении новых возможностей или модулей, которые поддерживаются в существующих выпусках, посредством тестирования только новой службы или возможности во взаимодействии с другими включенными службами. Необходимо также учитывать данные во время обновления программного обеспечения из-за нескольких критических ошибок, найденных в сети.

В следующих таблицах перечислены рекомендованные требования по обновлению для большей части корпоративных организаций с высоким коэффициентом готовности:

Запуск управления программным обеспечением	Требование к жизненному циклу программного обеспечения
Новый сетевой сервис. Например, новая магистраль ATM или новый Сервис VPN.	Завершенная проверка жизненного цикла ПО включая тестирование новой характеристики (в сочетании с другими включенными сервисами), проверка сжатой топологии, анализ производительности предполагаемой причины и тестирование профиля приложения.
Новые возможности сети не поддерживаются в	Завершенная проверка жизненного цикла ПО включая тестирование новой

выпуске текущего программного обеспечения. Примеры включают QoS и Многопротокольную коммутацию по меткам (MPLS).	характеристики, в сочетании с другими включенными сервисами, проверкой сжатой топологии, анализом производительности предполагаемой причины и тестированием профиля приложения.
Новая основная характеристика или модуль оборудования, который существует в текущем релизе. Например, добавляя новый модуль GigE, поддержку групповой адресации или DLSW.	Процесс управления кандидатами. Возможная полная проверка на основе требований для последнего релиза. Возможно ограниченное тестирование/проверка, если возможный управляющий процесс посчитает текущую версию потенциально допустимой.
Доработка дополнительной характеристики. Например, устройство TACACS для управления доступом.	Рассмотрите управление кандидатами на основе риска функции. Рассмотрите тестирование или макетирование новой характеристики на основе риска.
Программное обеспечение в производстве в течение двух лет или ежеквартального анализа программного обеспечения.	Управление кандидатами и решения на уровне предприятия относительно завершенного управления жизненным циклом для определения текущего приемлемого выпуска.

Экстренные обновления

В некоторых случаях организации сталкиваются с потребностью к обновлению ПО из-за неисправимых ошибок. Может привести к проблемам, если организация не имеет методологии непредвиденного обновления. Примерами проблем с программным обеспечением являются как неуправляемые обновления программного обеспечения (т.е. программное обеспечение обновляется без управления жизненным циклом программного обеспечения), так и ситуации, в которых сетевые устройства постоянно отказывают, но организация не может их обновить, поскольку процесс сертификации/тестирования следующей версии ПО еще не завершен. Cisco рекомендует процесс экстренного обновления для этих ситуаций, где ограниченная проверка и пилоты выполнены в меньшем количестве критически важных для бизнеса областей сети.

Если неисправимые ошибки происходят без явного метода обхода ошибки, и проблемой являются отнесенные ошибки ПО, Cisco рекомендует, чтобы поддержка Cisco была полностью занята, чтобы изолировать дефект и определить, если или когда исправление доступно. Если доступны исправления, Cisco рекомендует использовать цикл экстренного

обновления для быстрого определения проблем, которые можно устранить при ограниченном времени простоя. В большинстве случаев организация выполняет поддерживаемую версию кода, и проблемное исправление доступно в существующей более новой промежуточной версии программного обеспечения.

Организации могут также подготовиться для потенциальных срочных обновлений. Подготовка включает в себя миграцию на один из поддерживаемых релизов Cisco IOS и идентификацию/разработку версии-кандидата на замещение в рамках той же серии Cisco IOS, что и сертифицированная версия. Поддерживаемое программное обеспечение важно, так как это означает, что разработчики Cisco добавляют исправления ошибок к определенным версиям программного обеспечения. Путем поддержания поддерживаемого программного обеспечения в сети организация уменьшает сокращает время проверки достоверности из-за до ядра стабильного кода и более знакомого. Обычно замена кандидата производится на новый промежуточный образ программного обеспечения в рамках одной группы версий Cisco IOS без дополнительных функций и аппаратной поддержки. Если организация находится во впервые применившей фазе определенной версии ПО, стратегия замены кандидатов особенно важна.

Процесс получения сертификата

Процесс получения сертификата помогает гарантировать, что проверенное программное обеспечение последовательно развертывается в производственной среде организации. Шаги процесса получения сертификата должны включать идентификацию дорожки, определения цикла обновления, управление кандидатами, тестирование/проверку и некоторое использование пилотного производства. Простой процесс получения сертификата, однако, все еще помогает гарантировать, что непротиворечивые версии ПО развернуты в определенных дорожках.

Запустите процесс сертификации, определив пользователей из групп архитектуры, проектирования, развертывания, а также операционной группы, для составления плана проекта и управления процессом сертификации. Группа должна сначала полагать, что бизнес - цели и возможности ресурса гарантируют, что процесс получения сертификата будет иметь дальнейший успех. Затем, возложите общую ответственность частных лиц или групп за ключевые шаги в процесс получения сертификата включая управление дорожки, определения обновления жизненного цикла, тестирование/проверку и пилотов. Каждая из этих областей должна быть определена, утверждена, и формально передана в организации.

Также включайте рекомендации по качеству или утверждение в каждой фазе процесса получения сертификата. Это иногда называют качественным процессом логического элемента, потому что определенные критерии качества должны быть встречены, прежде чем процесс может переместиться в следующий шаг. Это способствует тому, что процесс сертификации является эффективным и подходящим для назначенных ресурсов. В целом, когда проблемы найдены с качеством в одной области, толчки процесса, усилие поддерживает один шаг.

Программные обеспечения кандидат могут не встретить определенные условия сертификации из-за качества программного обеспечения или неожиданного поведения. При обнаружении проблем, влияющих на среду, у организации должен быть дополнительный хорошо организованный процесс, чтобы сертифицировать более поздний промежуточный выпуск. Это помогает уменьшить требования к ресурсам и обычно эффективно, если организация может выяснить, что было изменено и какие дефекты были исправлены. Для организации весьма обычно испытать проблему с начальным кандидатом и

сертифицировать более поздний промежуточный Cisco IOS Release. Организации могут также сделать ограниченную сертификацию или предоставить предупреждения, если некоторые проблемы существуют и могут обновить к более позднему полностью сертифицированному выпуску, когда был проверен новый промежуточный период. Нижеприведенная блок-схема демонстрирует базовый процесс сертификации и включает систему контроля качества Quality Gates (проверка каждого блока):

Дизайн - выбор и проверка версий Cisco IOS

Наличие четко определенной методологии для выбора и проверки версий Cisco IOS помогает организации уменьшать простой простой из-за неуспешных попыток модернизации и незапланированных ошибок ПО.

Стадия проектирования включает управление кандидатами и тестирование/проверку. Управление кандидатами является процессом, используемым для определения определенных версий для определенных моделей ПО. Тестирование/проверка является частью процесса получения сертификата и гарантирует, что определенная версия программного обеспечения успешна в требуемой дорожке. Тестирование/проверка должно быть проведено в лабораторной обстановке со сжатой топологией и конфигурацией, подобной производственной среде.

Стратегия и инструменты выбора и проверки Cisco IOS

Каждая организация должна иметь процесс для выбора и проверки стандартных версий Cisco IOS для сети начиная с процесса для выбора версии Cisco IOS. Многопрофильная команда от архитектуры, разработки и операций должна определить и задокументировать процесс управления кандидатами. После того, как утвержденный, процесс должен быть передан соответствующей группе доставки. Также рекомендуется, чтобы стандартный шаблон управления кандидатами был создан, который может быть обновлен со сведениями о кандидате, поскольку это определено.

Не все организации имеют сложную лабораторную среду, которая может легко подражать производственной среде. Некоторые организации пропускают лабораторное испытание из-за расхода и способности вести новую версию в сети без главного воздействия на бизнес. Однако организации с высокой доступностью поощрены создать лабораторную работу, которая подражает рабочей сети и разработать тестирование/процесс проверки данных для обеспечения высокого тестового покрытия для новых ПО Cisco IOS версий. Организация должна позволить приблизительно шести месяцам создавать лабораторную работу. В это время организация должна работать для создания определенных планов тестирования и процессов, чтобы гарантировать, что лабораторная работа будет использоваться к ее полному преимуществу. Для Cisco IOS это означает создание определенных планов тестирования Cisco IOS относительно каждой дорожки необходимого программного обеспечения. Эти процессы являются ключевыми в более крупных организациях по причине того, что многие лаборатории не привыкли использовать новый продукты и пробные версии программного обеспечения.

В следующих разделах кратко описаны средства управления кандидатами и тестирования/проверки для использования в процессе выбора и проверки Cisco IOS.

Программные средства управления кандидатами

Примечание: Большинство средств, представленных ниже, могут использоваться только зарегистрированным пользователем, который вошел в систему.

- Комментарии к выпуску — Предоставляют сведения относительно аппаратных средств, модуля и поддержки характеристик выпуска. В процессе управления кандидатами необходимо просматривать комментарии к релизу— это позволяет проверить, поддерживается ли все нужное программное и аппаратное обеспечение в потенциальной версии, а также позволяет выявить все проблемы перехода, включая различное поведение по умолчанию и требования к обновлению.

Средства тестирования и проверки

Средства тестирования и проверки используются для сетевых решений с новым оборудованием, программным обеспечением и приложениями.

- **Генераторы трафика** — Генерируют потоки трафика по нескольким протоколам, и необработанные скорости передачи пакетов использовали моделировать скорость через какую-то конкретную ссылку, использующую определенные протоколы. Пользователи могут указать источник, адресат MAC и номера сокетов. Эти значения можно увеличивать на определенные шаги или конфигурировать, чтобы они были статическими / фиксированными, или чтобы они имели случайное приращение. Устройство формирования трафика обеспечивает формирование пакетов для следующих протоколов:IPМежсетевой пакетный обмен (IPX)DEC NetAppleСистемы XNSInternet Control Message Protocol (ICMP)Internet Group Management Protocol (IGMP)Обслуживание сети без установления соединения (CLNS)Протокол дейтаграммы пользователя (UDP)Виртуальная интегрированная сетевая служба (VINES)Пакеты канала передачи данныхПрограммные средства доступны от Agilent и Spirent Communication.
- **Счетчик пакетов/Capture/Decoder (Анализатор)** — Позволяет клиенту выборочно перехватывать и декодировать пакеты во всем пакете и уровнях канала передачи данных. Средство имеет возможность задания пользователем фильтров, что позволяет отбирать только указанных данных протокола. Фильтры далее позволяют пользователю задавать получение пакетов, совпадающих с определенным IP - адресом, номером порта или MAC-адресом. Эти средства выпускает компания Sniffer Technologies.
- **Симулятор сети / Эмулятор** — Позволяет клиенту заполнять таблицы маршрутизации определенных маршрутизаторов, на основе требований рабочей сети. Поддержка поколения маршрутов протокола IP RIP (Routing Information Protocol), OSPF, "промежуточная система - промежуточная система" (IS-IS), внутреннего протокола маршрутизации (IGRP), протокола Enhanced IGRP (EIGRP) и пограничного шлюзового протокола (BGP). Программные средства доступны от соединений с использованием потока пакетов и Spirent Communication.
- **Эмуляторы сеанса** — Генерируют потоки трафика по нескольким протоколам раздвижного окна и способны к передаче потоков трафика по нескольким протоколам через тестовую сеть к принимающему устройству. Принимающее устройство “отражает” пакеты назад к источнику. Источник проверяет количество отправленных, полученных, внеочередных и ошибочных пакетов. Это средство позволяет осуществлять гибкую настройку параметров окна в протоколе управления передачей (TCP), что дает возможность детально имитировать сеансы трафика клиент-сервер в лабораторной сети. Инструментарий доступен в Empirix.

- **Эмуляторы Крупномасштабной сети** — Справка в тестировании масштабируемости больших сред. Эти инструменты позволяют создавать и легко вводить в лабораторную топологию трафик управляющего типа, чтобы точнее имитировать рабочую среду. Возможности включают инжекторы маршрута, соседей по протоколу и соседей по протоколу Уровня 2. Программные средства доступны от Agilent и Spirent Communication.
- **Имитаторы WAN** — Идеал для тестирования трафика корпоративного приложения, где пропускная способность и задержка являются потенциально проблемой. Эти программные средства позволяют организациям локально тестировать приложение с предполагаемой задержкой и пропускной способностью, чтобы видеть как прикладные функции по глобальной сети (WAN). Эти инструменты часто используются для развертывания приложения и для профиля приложения тестового типа в рамках корпорации. Adtech, отдел коммуникаций Spirent и Shunra предоставляют программные средства моделирования глобальной сети (WAN).

Управление кандидатами

Управление кандидатами является процессом определения требований к версии программного обеспечения и потенциальных рисков для конкретного оборудования и включило наборы функций. Рекомендуется, чтобы организация провела четыре - восемь часов, должным образом исследующих требования к программному обеспечению, Комментарии к выпуску, ошибки ПО и потенциальные риски прежде, чем вести выпуск. Следующие структуры основание для управления кандидатами:

- Определите программные обеспечения кандидат через программные средства Cisco Connection Online (CCO).
- Завершенность ПО анализа риска, новая характеристика или поддержка кода.
- Определите и отследите известные ошибки в программном обеспечении, проблемы и требования в течение жизненного цикла.
- Определите поведение конфигурации по умолчанию выбранного изображения.
- Поддержите возврат и продвиньте вперед кандидатов на изменения потенциального кандидата.
- Кусты дефекта.
- Поддержка Расширенных сервисов Cisco.

Определение программных обеспечений кандидат стало более сложным с растущим числом продукции Cisco и версий ПО. CCO теперь имеет несколько программных средств включая планировщика обновления Cisco IOS, инструмент поиска программного обеспечения, матрицу совместимости оборудования программного обеспечения и средство обновления продукта, которое может помочь организациям определять потенциальных кандидатов на релиз. Эти программные средства могут быть найдены в http://www. cisco . com/cisco/software/navigator.html.

Затем, проанализируйте риск программного обеспечения потенциального кандидата. Это - процесс понимания, где программное обеспечение в настоящее время находится на кривой зрелости и затем взвешивании требований для развертывания с потенциальным риском предвыпускной версии. Например, если организация желает поместить программное обеспечение раннего развертывания (ED) в важную среду высокой доступности, связанный риск и потребность в ресурсах для успешной сертификации нужно рассмотреть. Организация должна, по крайней мере, добавить ресурсы для управления программным обеспечением для ситуаций с высокой степенью риска для обеспечения успеха. С другой

стороны, если версия общего развертывания (GD) доступна, который удовлетворяет потребности организации, тогда меньше ресурсов для управления программным обеспечением необходимо.

После определения потенциальных версий и рисков выполните поиск ошибок, чтобы определить критические ошибки, из-за которых может быть отказано в сертификации. Наблюдатель Дефекта Cisco, Навигатор Дефекта и Агенты Watcher Дефекта могут помочь определять потенциальные проблемы и должны использоваться всюду по жизненному циклу ПО для определения потенциальной угрозы безопасности или дефектных проблем.

Новое программное обеспечение кандидат должно также быть рассмотрено для потенциального поведения конфигурации по умолчанию. Это можно выполнить, просмотрев заметки о выпуске для нового образа программного обеспечения и проверив различия конфигурации с потенциальным образом, загруженным на предназначенные для этого платформы. Если выбранная версия не встречает условия сертификации в некоторый момент в процессе, управление кандидатами может также включать идентификацию версий возврата или перейти к версиям. Путем наблюдения дефектов, отнесенных к функциям указанной дорожки, организация может поддержать потенциальных кандидатов для сертификации.

Расширенные сервисы Cisco являются также превосходным программным средством для управления кандидатами. Эта группа может предоставить дальнейшее понимание в процесс разработки и совместную работу между большим числом отраслевых экспертов во многих других средах вертикального рынка. Обычно возможности лучших средств для устранения ошибок или управления кандидатами существуют при поддержке Cisco благодаря уровню экспертизы и знанию версий программного обеспечения, используемых в других организациях.

Тестирование и проверка

Тестирование и проверка является важным аспектом управления оптимальные методы и сеть с высоким уровнем доступности в целом. Правильное лабораторное тестирование может существенно снизить время простоя благодаря обучению сетевого поддерживающего персонала и помощи в оптимизации процесса реализации сети. Однако, чтобы обеспечить эффективность работы, организации следует выделять необходимые ресурсы для формирования и поддержки соответствующей лабораторной среды, использовать необходимые ресурсы для проведения правильных тестов, а также применять рекомендуемую методологию тестирования, которая включает в себя набор измерений. Без любой из этих областей тестирование и процесс проверки данных могут не оправдать надежды организации.

Большинство организаций не имеет рекомендуемой тестовой лабораторной среды. Поэтому много организаций имеют развернутые решения неправильно, испытали сбои при изменении сети или испытали неполадки программного обеспечения, которые, возможно, были изолированы в лабораторной среде. В некоторых средах это приемлемо, поскольку стоимость простоя не смещает стоимость сложной лабораторной среды. Во многих организациях, однако, не может быть допущено время простоя. Этим организациям необходимо срочно разработать рекомендуемые лабораторные тесты, типы тестов и методологию тестирования для улучшения качества производственной сети.

Лаборатория и среда тестирования

Лаборатория должна представлять собой изолированную зону с достаточным количеством места для столов, инструментов, тестируемых устройств, шкафов или полок с оборудованием. Самые крупные организации должны будут между четыре к оборудованию с десятью стойками подражать производственной среде. Некоторая физическая защита рекомендуется для поддержания тестовой среды во время тестирования. Это помогает препятствовать тому, чтобы лабораторное испытание было разрушено из-за других лабораторных приоритетов включая аппаратное заимствование, обучение или репетиции внедрения. Логической безопасности также рекомендуют препятствовать тому, чтобы поддельные маршруты ввели рабочую сеть или нежелательный трафик от выхода из лабораторной работы. Это можно сделать с помощью фильтров маршрутизации и расширенных списков доступа для лабораторного маршрутизатора шлюза. Подключение к рабочей сети полезно для загрузок программного обеспечения и доступа к лабораторной сети от производственной среды.

Лабораторная топология должна копировать производственную среду для всех конкретных тестовых планов. Воспроизводя аппаратные средства, топология сети и конфигурации функции рекомендуются. Конечно, репродуцирование действительной топологии почти невозможно, но что может быть сделано, должен воспроизвести сетевую иерархию и взаимодействие между устройствами производства. Это важно для протокольного или функционального взаимодействия между множественными устройствами. Некоторые тестовые топологии могут отличаться в зависимости от требований тестирования программного обеспечения. Граничное тестирование Cisco IOS глобальной сети (WAN), например, не должно требовать устройств типа LAN или тестирования и может только потребовать маршрутизаторы распределения глобальной сети (WAN) и Пограничные с WAN маршрутизатор. Ключ должен подражать функциональности программного обеспечения, не копируя производство. В некоторых случаях программные средства могут даже использоваться для имитации крупномасштабного поведения, такого как количество соседа по протоколу и таблицы маршрутизации.

Также необходимы средства для поддержки некоторых типов тестов путем улучшения возможности моделирования производственной среды и сбора тестовых данных. В число средств, используемых для имитации производства, входят сборщики трафика, генераторы трафика и устройства симулирования WAN. Smartbits служит хорошим примером устройства, которое может собирать и воспроизводить сетевой трафик или генерировать большие объемы трафика. Организация может также извлечь выгоду из устройств, которые могут помочь собирать данные, такие как анализаторы протокола.

Лабораторная работа также требует некоторого управления. Много более крупных организаций имеют полностью занятого специалиста лаборатории, который несет ответственность за управление лабораторной сетью. Другие организации используют для проверки лаборатории существующую архитектуру и технические службы. Ответственность управления лабораторной работы включает оборудование лабораторной работы заказа и отслеживание актива, кабельное подключение, управление физического пространства, определение правил лабораторной работы и направления, планирования лабораторной работы, лабораторной документации, устанавливание лабораторных топологий, запись планов тестирования, выполнение лабораторных испытаний, и управление потенциалом определило проблемы.

Типы проверок

Существуют различные типы тестов. Прежде, чем создать завершенную тестовую лабораторную работу и план тестирования, который может протестировать все в большом числе конфигураций, организация должна понять различные типы тестирования, намерение

тестирования, и должны ли разработка Cisco, технический маркетинг или отдел защиты интересов заказчиков или могли бы быть ответственны за некоторые различные тесты. Планы тестирования клиента обычно покрывают более представленные тестовые типы. Следующая таблица помогает понять различия в типах проверок, время их проведения и ответственные стороны.

Из тестов ниже, надлежащее тестирование определенного набора функций организации, топологии и смеси прикладных программ является обычно самым ценным. Важно знать, что Cisco выполняет полнофункциональный и регрессионное тестирование, однако Cisco не в состоянии протестировать профиль приложения вашей организации с вашей определенной комбинацией топологий, аппаратными средствами и настроенными функциями. Фактически, невозможно протестировать полный диапазон функций, аппаратных средств, модулей и перестановок топологии. Кроме того, Cisco не может протестировать совместимость со сторонним оборудованием. Cisco рекомендует, чтобы организации протестировали точную комбинацию аппаратных средств, модулей, функций и топологии, найденной в их среде. Это тестирование должно быть проведено в лабораторной работе, со свернутой топологией, представляющей производственную среду вашей организации с другими тестовыми типами поддержки, такими как производительность, совместимость, простой и выжигание дефектов.

Тест	Обзор теста	Ответственность за проведение теста
Функция и функциональность	Определяет, функционируют ли основные характеристики Cisco IOS и модули Оборудования CISCO, как объявлено. Функция или функциональные возможности модуля, а также параметры конфигурации функции должны быть протестированы. Удаление конфигурации и добавление должны быть протестированы. Стандартное тестирование в случае выхода из строя и	Тестирование устройства Cisco

	проведение тренировочных испытаний включены.	
Регрессия	Определяет, функционируют ли функция или модуль в сочетании с другими модулями и функциями, и если версия Cisco IOS функционирует в сочетании с другими версиями Cisco IOS относительно определенных функций. Включает некоторое выжигание дефектов и проверку бездействия системы.	Регрессионное тестирование Cisco
Производительность основных сведений об устройстве	Определяет базовую производительность функции или модуля, чтобы определить, удовлетворяют ли характеристика Cisco IOS или модули оборудования минимальные требования под загрузкой.	Тестирование устройства Cisco
Топология/Функция/Сочетание аппаратного обеспечения	Определяет, функционируют ли функции и модули как	Cisco тестирует объявленную топологию стандарта в лабораторных работах, таких как

	ожидалось в определенной топологии и модуле/функции/сочетании аппаратного обеспечения. **Тестирование должно включать проверку протокола, функций и команды show, испытание на принудительный отказ и простой.**	Enterprise Solutions Engineering (ESE) и разработка проверки интеграции сетевых решений (NSITE). Клиенты высокой доступности должны протестировать функцию/модуль/комбинации топологии как требуется, особенно с впервые применившим программным обеспечением и нестандартной топологией.
Выход из строя (анализ возможных вариантов)	Включает общие типы простоя или способы поведения, которые могут произойти в определенной функции/модуле/среде топологии и потенциальном влиянии функциональности. Тестирование на случай аварийного отключения включает замену плат, переключение каналов, моделирование сбоя устройств, каналов и плат.	Cisco ответственна за стандартное тестирование в случае выхода из строя. Клиенты в конечном счете ответственны за проблемы производительности простоя, отнесенные к масштабируемости их индивидуальной среды. Проверка бездействия системы должна быть сделана, если это возможно, в среде курса для клиента.
NetworkPerformance (Предполагаемая причина)	Исследует загрузку устройства относительно определенной	Клиенты в конечном счете ответственны за загрузку устройства и масштабируемость.

	функции/аппаратных средств/комбинации топологии. Акцент сделан на емкость и производительность устройств, например, на загрузку CPU, памяти, буфера и канала относительно установки типа трафика и требований к ресурсам для протоколов, окружения, числа маршрутов и других функциональных возможностей. Проверка помогает удостоверится в масштабируемости в больших средах.	Загрузка и проблемы масштабирования часто повышаются Представителями отдела продаж CISCO или Расширенными сервисами и часто тестируются с лабораториями Cisco, такими как Customer Proof-of-Concept Labs (CPOC).
Исправление ошибки	Гарантирует, что исправления ошибки восстанавливают определенный дефект.	Cisco тестирует исправления ошибки, чтобы гарантировать, что исправлена ошибка. Клиенты должны также протестировать, чтобы гарантировать, что ошибка, которую они испытали, исправлена и что дефект не ломает никакой другой

		аспект модуля или функции. Отладочные релизы проведены регрессионное тестирование, но промежуточныйрели зы обычно нет.
Управление сетью	Исследует возможности управления Протокола SNMP, переменную точность SNMP MIB, поддержку ловушек и Поддержку системного журнала.	Cisco ответственна за тестирование основных функций SNMP, функциональности и точности переменной MIB. Клиенты должны проверить Результаты управления сетью и в конечном счете ответственны за стратегию управления и методологию для развертываний новой технологии.
Эмуляция крупномасштаб ной сети	Эмуляция крупномасшта бной сети использует программные средства, такие как моделирующа я программа маршрутизато ра Agilent и комплект средств тестирования Спирана для моделирован ия больших сред. Это может включать соседей по протоколу, счетчики PVC (permanent virtual circuit) Frame Relay, размеры	Клиенты Cisco обычно ответственны за аспекты моделирования сети, тестирующего, который воспроизводит их сетевую среду, которая может включать количество соседей по протоколу маршрутизации / смежности и привязанные размеры таблицы маршрутизации и другие ресурсы, которые работают.

	таблиц маршрутизации, записи кэша и другие ресурсы, обычно необходимые в производстве, и которых нет в лаборатории по умолчанию.	
Совместимость	Если протокол или сигнальная совместимость требуются, тестирует все аспекты относительно подключения к стороннему сетевому оборудованию, особенно.	Клиенты Cisco обычно ответственны за все аспекты тестирования совместимости.
Выжигание дефектов	Исследует ресурсы маршрутизатора в течение долгого времени. Отбраковочные испытания, как правило, требуют, чтобы устройство находилось под некоторой загрузкой с расследованием использования ресурса включая память, ЦП и буферы в течение	Cisco выполняет основное проведение тренировочных испытаний. Клиентская проверка рекомендуется относительно уникальной топологии, устройства и комбинаций признаков.

	долгого времени.	

Методология тестирования

Как только организация знает то, что они тестируют, методология должна быть разработана для процесса тестирования. Цель оптимальной методологии тестирования – гарантировать, что данные, согласованные во время тестирования, всеобъемлющи, правильно задокументированы, легко воспроизводимы и ценны с точки зрения поиска потенциальных рабочих проблем. Лабораторные сценарии документации и воссоздания особенно важны для тестирования более поздних версий или для тестирования исправлений ошибки, найденных в лабораторной среде. Шаги методологии тестирования показывают ниже. Некоторые этапы проверки также можно выполнять одновременно.

1. Создайте проверку топологии, которая моделирует производственную среду под тестом. Граничная тестовая среда глобальной сети (WAN) может включать несколько центральных маршрутизаторов и один краевой маршрутизатор только, в то время как тест LAN может включать больше устройств, которые могут лучше всего представлять среду.
2. Настройте функции, которые моделируют производственную среду. Конфигурация устройств лабораторной работы должна близко совпасть с ожидаемыми конфигурациями программного и аппаратного обеспечения устройств производства.
3. Запишите план тестирования, определив тесты и цели, документируя топологию, и определив функциональные тесты. Тесты включают проверку основного протокола, проверку команды show, проверку нарушения связи и испытание на принудительный отказ. Пример определенного теста в плане тестирования найден в следующей таблице.
4. Проверьте маршрутизацию и функции протокола. Документ или срок ожидали результаты **команды показа**. Протоколы должны включать такие протоколы второго уровня, как ATM, Frame-Relay, Cisco Discovery Protocol (CDP), Ethernet и Spanning-Tree, а также такие протоколы третьего уровня, как IPX и многоадресный.
5. Проверьте работоспособность функции. Документ или срок ожидали результаты **команды показа**. Функции могут включать команды глобальной кофигурации и любые критически важные возможности, такие как аутентификация, авторизация и учет (AAA).
6. Моделируйте нагрузку, которая ожидается в производственной среде. Моделирование загрузки может быть сделано с коллекторами трафика / генераторы. Проверьте ожидаемые переменные использования сетевого устройства, включая CPU, память, использование буфера и статистические данные интерфейса с изучением потери пакетов. Документ или срок ожидали результаты **команды показа**.
7. Выполните проверку бездействия системы, где устройство и программное обеспечение, как ожидали бы, будут иметь дело с или предотвращать под загрузкой. Например, извлечение карты, колебания связи, колебание маршрута и широковещательные штормы. Гарантируйте, что корректные trap-сообщения SNMP генерируются на основе функций, используемых в сети.
8. Результаты тестирования документа и измерения устройства как тесты должны быть повторимыми.

Тестовое название	**Аварийное переключение протокола HSRP**
Тестовые	Примените загрузку к интерфейсу

конфигурационные требования	основного шлюза. Приблизительно 20% трафика должно направляться к шлюзу от абонентской станции, а 60% входящего трафика - в направлении абонентской станции. Кроме того, увеличьте трафик до более высокой загрузки.
Шаги теста	STP монитора и HSRP через **команды показа**. Откажите подключение интерфейса основного шлюза и затем восстановите соединение после того, как будет собрана информация.
Ожидаемые измерения	ЦП во время аварийного переключения. Отображать интерфейс для основного и дополнительного шлюзов до, во время и после отказа. Вид HSRP до, во время и после.
Ожидаемые результаты	Основной шлюз при отказе переключается на шлюз другого маршрутизатора в течение двух секунд. **команды показа** должным образом отражают изменение. Когда подключение восстановлено, аварийное переключение к основному шлюзу происходит.
Фактические результаты	
Пройдено (Pass) или ошибка (Fail)	
Модификации, требуемые для достижения прохода	

Измерения устройства

Во время этапа тестирования выполните и задокументируйте следующие измерения, чтобы гарантировать, что устройство выполняет правильно:

- Использование памяти
- Загрузки ЦПУ
- Использование буфера
- Статистика интерфейса
- Таблицы маршрутов
- Определенная отладка

Данные для измерений зависят от выполняемого теста. Может также быть дополнительная

информация для измерения, в зависимости от конкретных решаемых вопросов.

Для каждого приложения, которое тестируется, параметры меры для обеспечения, на данном приложении нет никакого неблагоприятного влияния на производительность. Это завершено путем использования базовой производительности, которая может использоваться для сравнения производительности пред и почтовых развертываний. Примеры для измерительных тестов приложения включают:

- Среднее время, требуемое для регистрации в сети.
- Среднее время, необходимое сетевой файловой системе (NFS) для копирования группы файлов.
- Среднее время это берет, чтобы запустить приложение и быть предложенным с первым экраном.
- Другие параметры, зависящие от приложения.

Реализация - быстрые и успешные развертывания Cisco IOS

Четко определенный процесс внедрения позволяет организации эффективно развертывать новые ПО Cisco IOS версии.

Фаза реализации включает пилотный процесс и процесс внедрения. Пилотный процесс гарантирует, что версия Cisco IOS будет успешна в среде, и процесс внедрения позволяет быстрые и успешные развертывания Cisco IOS более широкого масштаба.

Стратегия и инструменты развертывания Cisco IOS

Стратегия развертывания Cisco IOS заключается в проведении окончательной сертификации при помощи пилотного процесса и быстрого развертывания, используя средства обновления и четко определенный процесс внедрения.

Прежде, чем инициировать сетевой пилотный процесс, много организаций создают общие экспериментальные рекомендации. Руководство по тестовому внедрению должно включать все ожидаемые исходы такого внедрения, такие, как критерии успешного исхода, допустимое размещение опытного образца, документацию, ожидания владельца опытного образца, требования уведомления пользователя и ожидаемая длительность внедрения. Многопрофильная команда от разработки, реализации и операций обычно вовлекается в построение полных экспериментальных рекомендаций и пилотного процесса. Как только будет создан процесс проводника, отдельные группы реализации смогут руководить успешными проводниками в обычном режиме, используя наиболее эффективные методы.

После утверждения развертывания и окончательной сертификации новой версии программного обеспечения организации необходимо начать планирование обновления Cisco IOS. Планирование начинается с определения требований для нового образа, включая платформу, память, флэш-память и конфигурацию. Группы архитектуры и инженерные группы обычно определяют требования нового образа ПО в фазе управления кандидатами жизненного цикла управления Cisco IOS. Как только требования определены, каждое устройство должно быть проверено и по возможности обновлено группой внедрения. Модуль диспетчера образов ПО CiscoWorks2000 (SWIM) может также проходить этап проверки, подтверждая соответствие требований Cisco IOS к управлению оборудованием. Если все устройства проверены и обновлены до правильных стандартов с новыми образами, то группа реализации начинает процесс реализации с помощью

медленной загрузки, используя SWIM-модуль CiscoWorks2000 в качестве средства для развертывания программного обеспечения.

После нескольких успешных попыток развертывания нового образа организация может начать быстрое развертывание с использованием CiscoWorks SWIM.

Управление запасами Cisco IOS

Диспетчер инвентаризации основных инструментов диспетчера ресурсов (RME) в CiscoWorks2000 значительно упрощает управление версиями маршрутизаторов и коммутаторов Cisco через веб-инструменты создания отчетов, которые сообщают об устройствах Cisco IOS и сортируют их на основе версии программного обеспечения, платформы и имени устройства.

SWIM Cisco IOS

SWIM CiscoWorks2000 может помочь в сокращении подверженных ошибкам сложностей процесса обновления. Встроенные ссылки на CCO коррелируют Cisco онлайновая информация об исправлениях программного обеспечения с Cisco IOS и Программным обеспечением Catalyst, развернутым в сети, выделяя отнесенные технические примечания. Когда модернизации оборудования (ПЗУ начальной загрузки, ОЗУ Флэша) необходимы для поддержки предложенных обновлений образа программного обеспечения, новые программные средства планирования находят системные требования и передают уведомления.

Прежде чем обновление инициируется, предварительные условия нового образа проверены против целевого коммутатора или данных учета маршрутизатора, чтобы помочь гарантировать успешное обновление. При обновлении нескольких устройств SWIM синхронизирует задачи загрузки и позволяет пользователю контролировать ход выполнения задания. Запланированными заданиями можно управлять с помощью процесса выхода из системы, предоставляя руководителям возможность утверждать действия специалистов перед инициализацией задания обновления. RME 3.3 имеет возможность анализировать обновления программного обеспечения для платформ Cisco IGX, BPX и MGX, что значительно упрощает работу и сокращает время, необходимое на определение того, нужно ли обновить существующее программное обеспечение.

Пилотный процесс

Для уменьшения потенциальной незащищенности и более безопасно перехватывать любые остающиеся проблемы производства, пробный релиз программы рекомендуется. Обычно пилоты важнее при развертывании новых технологий, однако внедрение многих новых программ будет связано с новыми службами, возможностями или аппаратным обеспечением, где пилот более важен. Индивидуальный пилотный план должен включать выбор пилотного плана, его длительность и показатели. Выбор пробной версии – это процесс определения, когда и где должна выполняться пробная версия. Пробное измерение – это процесс сбора необходимых данных для определения успехов, сбоев и возможных проблем.

Экспериментальный выбор определяет, где и как будет завершен пилот. Пилот может запустить с одного устройства в низкой зоне поражения и расшириться на составные устройства в более высокой зоне поражения. Некоторые соображения для выбора пилота, когда влияние может уменьшиться:

- Установленный в области сети, эластичной к одиночному устройству, влияют из-за резервирования.
- В области сети с минимальным количеством пользователей позади выбранного устройства, кто может иметь дело с некоторым возможным производственным влиянием.
- Рассмотрите разделение пилота вдоль линий архитектуры. Например, ведите его на доступе, распределении и/или магистральных уровнях сети.

Продолжительность тестового внедрения рассчитывается исходя из времени, которое требуется для проведения полного тестирования и оценки всех функций устройств. Это должно включать и выжигание дефектов и сеть под загрузками обычного трафика. Длительность также зависит от этапа обновления кода и области сети, Cisco IOS. Если Cisco IOS является новым основным релизом, более длинный период контроля предпочтен. Так как обновление представляет собой отладочную версию с минимальным набором новых функций, достаточно более короткого периода контроля.

Во время пилотной фазы важно контролировать и задокументировать результаты подобным образом как начальное тестирование. Это могут быть опросы пользователей, сбор пилотных данных, сбор данных о неполадках и критерии успешности/безуспешности выполнения. Частные лица должны быть непосредственно ответственны за отслеживание и мониторинг экспериментального выполнения, чтобы гарантировать, что все проблемы определены и это, пользователи и сервисы, вовлеченные в пилота, удовлетворены экспериментальными результатами. Большинство организаций будет сертифицировать выпуск, если это будет успешно в пилоте или производственной среде. Этот шаг – критический опасный отказ в некоторых средах из-за воспринимаемой успешности, когда критерии измерения или успешности не определены и не документированы.

Реализация

После того, как пилотная фаза была завершена в рабочей сети, начните фазу реализации Cisco IOS. Этап реализации предусматривает несколько действий по обеспечению успешного обновления программного обеспечения и эффективной реализации, включая постепенную реализацию, итоговую сертификацию, подготовку к обновлению, автоматизацию обновления и итоговую проверку.

Slowstart реализации является процессом медленной реализации недавно протестированного выпуска, чтобы гарантировать, что образ имеет полное воздействие производственной среды перед преобразованием полного масштаба и окончательной сертификацией. Некоторые организации могут начать с одного устройства и одного дня демонстрации до обновления двух устройств на следующий день и возможно несколько дней. Когда приблизительно устройства TEN были размещены в производство, организация может дождаться к одной - двум неделям перед окончательной сертификацией определенной версии Cisco IOS. Полная сертификация позволяет обеспечить более быстрое развертывание определенной версии при более высоком уровне надежности.

После процесса медленного пуска все устройства, определенные для обновления, должны быть рассмотрены и проверили использование сведений об устройствах и матрицы стандартов минимального номера версии Cisco IOS для начальной загрузки, DRAM и флэш-памяти, чтобы гарантировать, что удовлетворены требования. Данные можно получить через внутренние инструменты, сторонние инструменты SNMP или через использование CiscoWorks2000 RME. Перед реализацией устройство CiscoWorks2000 SWIM осуществляет проверку и анализ этих переменных. Однако это всегда - хорошая идея знать, что ожидать

во время попыток реализации.

Если больше, чем аналогичные устройства сто планируются для обновлений, строго рекомендуется, чтобы был использован автоматический метод. Автоматизация, как показывали, повысила эффективность обновления и улучшила процент от успехов обновления устройств во время больших развертываний, на основе внутреннего обновления 1000 устройств с и без SWIM. Cisco рекомендует, чтобы SWIM CiscoWorks 2000 использовался для большого развертывания к уровню проверки, который выполнен во время обновления. Если проблема будет обнаружена, SWIM даже отступит из версии Cisco IOS. SWIM функционирует путем создания и планирования заданий обновления, где задание настроено с устройствами, желаемыми образами обновления, и время выполнения задания. Каждое задание должно содержать двенадцать или меньше обновлений устройств, и до двенадцати заданий могут работать одновременно. Кроме того, после планового обновления SWIM проверяет, успешно ли запускается новая версия Cisco IOS. Рекомендуется позволить приблизительно двадцать минут для каждого обновления устройств (включая проверку). Использование этой формулы, организация может обновить тридцать шесть одних устройств в час. Cisco также рекомендует, чтобы максимум устройств сто был обновлен в вечер для сокращения воздействия потенциальной проблемы.

После автоматизированного обновления некоторая проверка должна быть сделана для обеспечения успеха. Средство CiscoWorks2000 SWIM может запускать настраиваемые сценарии после обновления для дополнительной проверки успешного обновления. Процедура верификации включает в себя определение наличия нужного количества маршрутов на маршрутизаторе, проверку активности и рабочего состояния интерфейсов или доступности устройства. Следующий типовой чек-лист может полностью проверить успех развертываний Cisco IOS:

- Устройство должным образом перезагружалось?
- Доступность устройства для эхо-теста и достижима через платформы системы управления сетью (NMS)?
- Ожидаемые интерфейсы на устройстве и активны?
- Устройство имеет корректные смежности протокола маршрутизации?
- Таблица маршрутизации заполнена?
- Проходящий трафик устройства правильно?

Эксплуатация – управление широким доступом Cisco IOS реализаций

Использование оптимального метода высокой доступности среды Cisco IOS помогает уменьшать сложность сети, улучшать время устранения проблемы и улучшать доступность сети. Раздел эксплуатации документа по управлению Cisco IOS включает стратегии, средства и наилучшие методы, рекомендуемые для управления Cisco IOS.

Оптимальные методы для операций Cisco IOS включают управление версиями программного обеспечения, Управление системным журналом Cisco IOS, управление проблемами, стандартизацию конфигурации и управление доступностью. Управление версиями программного обеспечения является процессом отслеживания, проверки и улучшения непротиворечивости программного обеспечения в определенных моделях ПО. Управление системным журналом Cisco IOS является процессом упреждающего мониторинга и реакции на Сообщения системного журнала более высокого приоритета, генерируемые Cisco IOS. Принцип управления проблемами заключается в быстром и

эффективном сборе критически важной информации по проблемам, связанным с программным обеспечением, в целях предотвращения дальнейшего их возникновения. Стандартизация конфигурации является процессом стандартизации конфигураций для сокращения потенциала для непротестированного кода, который будет осуществлен в производстве и стандартизирует поведение функции и сетевой протокол. Управление доступностью является процессом улучшающейся доступности на основе метрик, целей усовершенствования и проектов улучшения.

Стратегии и инструменты управления Cisco IOS

Много стратегий качества и программных средств существуют, чтобы помочь управлять средами Cisco IOS. Основная ключевая стратегия деятельности Cisco IOS состоит в том, чтобы сохранять как можно более простую среду, избегая изменений в конфигурации и версиях Cisco IOS, насколько это возможно. Сертификация Cisco IOS была уже обсуждена, однако единообразие конфигурации является другой ключевой областью. Архитектурно-техническая группа должна заниматься разработкой стандартов конфигурации. Группа внедрения и оперативная группа несут ответственность за настройку и поддержание стандартов с помощью контроля версии Cisco IOS и стандартов / контроля конфигурации.

Вторая стратегия операций Cisco IOS является способностью определить и быстро решить сбои сети. Проблемы сети должны обычно определяться группой операций, прежде чем пользователи призовут их. Все проблемы необходимо устранять как можно быстрее, чтобы они не успевали затронуть или изменить рабочую среду. Несколько ключевых лучших методов в этой области являются Управлением системным журналом Cisco IOS и управлением проблемами. Программное средство, чтобы помочь быстро диагностировать сбои программного обеспечения Cisco IOS является Интерпретатором выходных данных Cisco.

Третья стратегия является постоянным усовершенствованием. Первичный процесс должен улучшить основанную на качестве программу усовершенствования доступности. Путем выполнения анализа корневых причин по всем проблемам, включая связанные проблемы Cisco IOS, организация может улучшить тестовое покрытие, улучшить времена устранения проблемы и улучшить процессы, которые устраняют или уменьшают влияние простоя. Кроме того, организация может обнаруживать общие проблемы и создавать процессы для их быстрого устранения.

В Cisco IOS предусмотрены такие средства управления, как инструментальные средства управления версиями программного обеспечения (CiscoWorks2000 RME), управление системным журналом и диспетчеры конфигурации устройства, обеспечивающие управление согласованностью конфигураций устройств.

Управление системным журналом

Сообщения системного журнала – это сообщения, посланные устройством на собирающий сервер. Сообщения могут уведомлять об ошибках (например, об отключении канала) или содержать информацию о том, что некий пользователь выполнил вход для настройки терминала на устройстве.

Журнал программных средств управления системным журналом и Сообщения системного журнала дорожки, полученные маршрутизаторами и коммутаторами. Некоторые инструменты снабжены фильтрами, позволяющими удалять нежелательные сообщения, которые могут отвлекать от важных сообщений. Инструменты системного журнала должны

также разрешать создание отчетов на основе полученных сообщений. Отчеты можно просматривать за определенный временной период, для отдельных устройств, типов или приоритетов сообщений.

Самое популярное программное средство Системного журнала для управления Cisco IOS является диспетчером системного журнала RME CiscoWorks2000. Другие программные средства доступны включая SL4NT, условно-бесплатную программу от Netal и Private я от OpenSystems.

Менеджер конфигурации устройства CiscoWorks

Менеджер Конфигурации устройства CiscoWorks2000 поддерживает активный архив и предоставляет простой способ для обновления изменений конфигурации через множественные маршрутизаторы Cisco и коммутаторы. Когда изменение обнаружено и регистрирует изменение информации к Сервису Изменения данных аудита, менеджер конфигурации контролирует сеть для изменений конфигурации, обновляет архив. Веб-интерфейс пользователя позволяет вам искать архив определенные атрибуты конфигурации и сравнивать содержание двух файлов конфигурации для простой идентификации различий.

Интерпретатор выходных данных Cisco

Интерпретатором выходных данных Cisco – это инструмент, используемый для диагностики аварийных ситуаций, вызванных программным обеспечением. Средство может помочь в определении дефектов программного обеспечения без необходимости звонка в Центр технической поддержки Cisco (TAC), или оно может быть использовано в качестве основных сведений для отправки в TAC вместе с вызванным программным обеспечением сбоем. Эта информация будет обычно помогать ускорять разрешение к проблеме, по крайней мере с точки зрения набора необходимой информации.

Контроль версий программного обеспечения

Контроль версий программного обеспечения - это процесс внедрения только стандартных версий программного обеспечения с контролем сети для выявления и возможного изменения программного обеспечения в связи с несовместимостью версий. В целом управление версиями программного обеспечения выполнено с помощью контроля за стандартами и процесса получения сертификата. Много организаций публикуют стандарты версии на центральном Web-сервере. Кроме того, штат реализации обучен рассмотреть то, что версия выполняет и обновить версию, если это не совместимый со стандартами. Некоторые организации имеют качественный процесс логического элемента, где второе подтверждение завершено посредством аудитов, чтобы гарантировать, что стандарт придерживается во время реализации.

Если сеть и штат операций являются большими, во время операции весьма распространено видеть нестандартные версии в сети, особенно. Это может происходить из-за необученности нового персонала, ненастроенных загрузочных команд или непроверенной реализации. Это всегда - хорошая идея периодически проверить программные средства использования стандартов версии ПО, такие как RME CiscoWorks 2000, который может сортировать все устройства версией Cisco IOS. Когда обнаруживаются несоответствия стандартам, они должны быть немедленно помечены, сформированы уведомления о неисправности или изменении для доведения версии до определенного стандарта.

Упреждающее управление системным журналом

Коллекции системного журнала, контроля и анализа являются процессами управления ошибками, которые рекомендуется использовать для разрешения специфических для сети Cisco IOS проблем, которые сложно или невозможно обнаружить другими способами. Коллекция системного журнала, мониторинг и анализ помогают улучшать время устранения проблемы путем определения и устранения многих ошибок заранее, прежде чем большие серьезные сетевые проблемы испытают или сообщат пользователи. Системный журнал также предоставляет более эффективный метод сбора большого разнообразия проблем когда по сравнению с последовательным Последовательным опросом SNMP для большого числа переменных MIB. Коллекция системного журнала, мониторинг и анализ выполнены путем использования корректной Конфигурации Cisco IOS, программных средств корреляции Системного журнала, таких как RME CiscoWorks2000 и/или управление События системного журнала (syslog). Управление события системного журнала (syslog) сделано путем парсинга собранных Данных системного журнала для определенных критических сообщений и затем пересылки предупреждения или trap-сообщения Диспетчеру событий для уведомления в реальном времени и разрешения.

Контролирование системного журнала требует поддержки инструментов NMS или сценария для анализа и отчета о данных системного журнала. Это включает возможность сортировки сообщений системного журнала по дате или времени, устройству, типу сообщения или частоте сообщений. В больших сетях программные средства или сценарии могут быть внедрены, чтобы проанализировать Данные системного журнала и передать предупреждения или уведомления системам управления событиями или операциям и техническому персоналу. Если предупреждения для большого разнообразия Данных системного журнала не используются, организация должна рассматривать Данные системного журнала более высокого приоритета, по крайней мере, ежедневно и создавать ярлыки проблемы для потенциальных проблем. Для упреждающего обнаружения проблем сети, которые не могут быть замечены посредством обычного мониторинга, периодическое изучение и анализ данных истории системного журнала должны быть выполнены для обнаружения ситуаций, которые могут не указать на неотложную проблему, но могут предоставить индикацию относительно проблемы, прежде чем это станет сервисным влиянием.

Решение проблем

Много клиентов испытывают дополнительное время простоя в связи с к отсутствию процессов в управлении проблемами. Когда администраторы сети пытаются решить проблему быстро с помощью комбинации влияющих на сервис команд или изменений конфигурации вместо того, чтобы провести время на распознавании ошибки, сборе сведений и хорошо проанализированном пути решения, дополнительное время простоя может произойти. Наблюдаемое состояние в этой области включает перезагружающиеся устройства или очищающиеся таблицы IP-маршрутизации прежде, чем исследовать проблему и ее основную причину. В некоторых случаях это происходит из-за целей устранения проблемы поддержки первого уровня. Главная задача устранения всех программных ошибок должна заключаться в оперативном сборе необходимой информации для анализа причины проблемы перед тем, как восстанавливать соединение или работу службы.

Процесс управления проблемами рекомендуется в больших средах. **Прежде чем перейти на следующий уровень, необходимо иметь определенное описание стандартных проблем и результаты выполнения соответствующих show-команд** . Первая поддержка уровня никогда

не должна очищать маршруты или повторно загружать устройства. Организация первого уровня должна быстро собирать сведения и переходить на второй уровень. Путем расходов еще всего нескольких минут первоначально на распознавание ошибки или описание проблемы, обнаружение rootcause намного более вероятно, таким образом позволяет обходной путь, лабораторную идентификацию и создание отчетов дефекта. Поддержка второго уровня должна быть хорошо сведущей в типах информации, в которой, возможно, нуждается Cisco, чтобы диагностировать проблему или подать отчет об ошибках. **Это включает в себя дампы памяти, вывод данных маршрутизации и вывод команды device show.**

Стандартизация конфигурации

Стандарты конфигурации глобального устройства представляют практику поддержания стандартных параметров глобальной конфигурации через подобные устройства и сервисы, приводящие к предприятию широкая согласованность глобальной конфигурации. Команды глобальной кофигурации являются командами, которые применяются ко всему устройству а не к отдельным портам, протоколам или интерфейсам. Команды глобальной кофигурации обычно влияют на доступ к устройству, общее поведение устройства и безопасность устройства. В Cisco IOS это включает сервисные команды, команды IP, команды VTY, команды консольного порта, регистрируя команды, команды AAA/TACACS +, команды SNMP и команды banner. Также важный в стандартах конфигурации глобального устройства соответствующее соглашение о записи имен устройства, которое позволяет администраторам определять устройство, тип устройства и размещение устройства на основе Имени системы доменных имен (DNS) устройства. Согласованность глобальной конфигурации важна для общего уровня обслуживаемости и надежности сетевой среды, потому что это помогает уменьшать сложность сети и улучшать наличие поддержки сети. Трудности с поддержкой часто происходят без настройки стандартизации вследствие некорректного или несогласованного поведения устройства, доступа SNMP и общей безопасности устройства.

Поддержание стандартов конфигурации глобального устройства обычно выполняется внутренней разработкой или группой операций, которая создает и поддерживает параметры глобальной конфигурации для однородных сетевых устройств. Это - также полезный прием для обеспечения копии файла глобальной конфигурации в каталогах TFTP так, чтобы они могли быть первоначально загружены ко всем недавно обеспеченным устройствам. Также полезный веб-доступный файл, который предоставляет файлу стандартной конфигурации пояснение каждого параметра конфигурации. Некоторые организации даже глобально настраивают подобные устройства на периодической основе, чтобы помочь обеспечению согласованности глобальной конфигурации, или периодически просматривают устройства по стандартам правильной глобальной конфигурации. Стандарты конфигурации протоколов и интерфейсов отражают практику поддержания стандартов конфигураций для интерфейсов и протоколов.

Слаженность конфигураций протокола и интерфейса улучшает доступность сети благодаря уменьшению сложности сети, предоставления ожидаемого поведения устройства и протокола и улучшая возможность поддержки сети. Несовместимость конфигурации протокола или интерфейса может вызвать неожиданное поведение устройства, проблемы маршрутизации трафика, возросшие проблемы соединения, и возросшее время реагирующей поддержки. Стандарты конфигурации интерфейса должны включать дескрипторы интерфейса CDP, конфигурацию кэширования и другой протокол определенные стандарты. Протокол определенные стандарты конфигурации может включать:

- Конфигурация IP-маршрутизации
- Конфигурация DLSw
- Конфигурация списка доступа
- Конфигурация ATM
- Конфигурация Frame Relay
- Конфигурация связующего дерева
- Назначение VLAN и конфигурация
- Протокол виртуального транкинга (VTP)
- HSRP

Примечание: Возможно иметь другой протокол определенные стандарты конфигурации в зависимости от того, что настроено в сети.

Пример стандартов IP может включать:

- Размер подсети
- Пространство IP-адресов используется
- Используемый протокол маршрутизации
- Конфигурация протокола маршрутизации

Поддержание протокола и стандартов конфигурации интерфейса обычно является ответственностью проектирования сети и групп реализации. Инженерная группа должна отвечать за определение, тестирование, проверку и документирование стандартов. Группа реализации тогда ответственна за использование конструкторских документаций или шаблонов конфигурации для инициализации новых сервисов. Инженерной группе следует создать документацию для всех аспектов необходимых стандартов для обеспечения целостности. Шаблоны конфигурации должны также быть созданы, чтобы помочь принуждать стандарты конфигурации. Рабочие группы также следует обучить стандартам, и они должны быть способны выявлять проблемы нестандартной конфигурации. Единообразие конфигурации имеет большую помощь в тестировании, проверке и фазе сертификации. Фактически, без стандартизированных шаблонов конфигурации, почти невозможно соответственно протестировать, проверить, или сертифицировать версию Cisco IOS для умеренно большая сеть.

Управление доступностью

Управление доступностью является процессом повышения качества с помощью доступности сети в качестве метрики повышения качества. Много организаций теперь измеряют тип простоя и доступность. Типы выхода из строя могут включать оборудование, программное обеспечение, канал и носитель, электропитание и среду, а также пользовательские ошибки или процессы. Путем определения простоев и выполнения анализа корневых причин сразу после восстановления, организация может определить методы для улучшения доступности. Почти все сети, которые достигли высокой доступности, имеют некоторый процесс повышения качества.

Приложение A - версии обзора Cisco IOS

Стратегия опубликования Cisco IOS software release построена на тщательной разработке ПО, обеспечении качества и быстрого выхода на рынок, что совершенно необходимо для успешной работы сетей клиентов Cisco.

Процесс определяется четырьмя категориями версий, которые описаны ниже:

- Релиз раннего развертывания (ED)
- Основной релиз
- Релиз с ограниченным применением (LD)
- Релиз для общего развертывания (GD)

Cisco создает и поддерживает новую версию IOS, которая имеет информацию об отдельных версиях, целевых рынках, способах перехода, описаниях новых характеристик, и так далее.

На рисунке, приведенном ниже, показан жизненный цикл Cisco IOS software release:

Релизы для первоначального развертывания

Релизы для первоначального развертывания Cisco IOS являются механизмами, которые приносят новую разработку к рынку. Каждая отладочная версия Релиза для первоначального развертывания включает не только исправления ошибки, но также и ряд новых характеристик, новой поддержки платформ и главных улучшений к протоколам и Инфраструктуре Cisco IOS. Все к двум годам, функциям и платформам Релизов для первоначального развертывания портируются на следующий Cisco IOS Release магистрали.

Существует четыре типа выпусков ED, каждый из которых отличается моделью выпуска и контрольными датами жизненного цикла. Релизы для первоначального развертывания могут быть классифицированы как:

- **Версии Consolidated Technology Early Deployment (CTED)** — новая ПО Cisco IOS модель релизов использует объединенный обучающий курс по релизу ED, также известный как серия "T", для представления новых характеристик, новых аппаратных платформ и других усовершенствований к Cisco IOS. Их называют совместной технологией, потому что они превышают внутренние Служебные подразделения (BU) и определения Направления деятельности (LOB). Примерами объединенных технологических релизов является Cisco IOS 11.3t, 12.0T, и 12.1T.
- **Версии Specific Technology Early Deployment (STED)** — Релизы STED имеют подобные характеристики обязательств по функции как Релизы CTED за исключением того, что они предназначаются для определенной технологии или театра рынка. Они всегда выпускаются для конкретных платформ и только под управлением Cisco BU. Выпуски STED идентифицируются по двум буквам, добавленным к основной версии выпуска. Примерами Релизов STED является Cisco IOS 11.3NA, 11.3MA, 11.3WA, и 12.0 дальтонов.
- **Версии Specific Market Early Deployment (SMED)** — S ED Cisco IOS дифференцируются от STED фактом, что они предназначаются для определенного сегмента вертикального рынка (интернет-провайдеры, предприятия, финансовые учреждения, компании Telcom, и так далее). S ED включают определенные требования к характеристикам технологии только для определенных платформ релевантности, используемых намеченным вертикальным рынком. Они могут дифференцироваться от CTED фактом, что они только созданы для определенных платформ релевантности к вертикальному рынку, тогда как CTED были бы созданы для большего количества платформ на основе более широких технологических требований. Версии S ED Cisco IOS определены одной буквой, добавленной к основной версии релиза (точно так же, как CTED). Примерами S ED является Cisco IOS 12.0S и 12.1E.
- **Недолгие Релизы раннего развертывания, также известные как X Версий (XED)** — версии CISCO IOS XED, представляют новые аппаратные средства и технологии на

рынок. Они не предоставляют ни версий поддержки, ни промежуточных версий программного обеспечения. Если дефект найден в XED до его конвергенции с CTED, переделка ПО инициируется, и номер добавлен к названию. Например, Cisco IOS Release 12.0 (2) XB1 и 12.0 (2) XB2 являются примерами 12.0 (2), XB восстанавливает.

Основные релизы

Основные релизы являются основными средствами развертывания для продуктов программного обеспечения Cisco IOS. Они находятся под контролем отдела технологий Cisco IOS и объединяют в себе функции, платформы, функциональные возможности, технологии и распространение хостов из предыдущих выпусков ED. Основные релизы Cisco IOS ищут большую устойчивость и качество. По этой причине основные релизы не принимают добавление функций или платформ. Каждая отладочная версия предоставляет исправления ошибки только. Например, Cisco IOS Software Release 12.1 и 12.2 являются основными релизами.

Основные релизы имеют плановые обновления обслуживания, названные отладочными релизами, которые полностью проведены регрессионное тестирование, включают новые исправления ошибки и не поддерживают новых платформ или функций. По номеру выпуска можно определить основной выпуск и его уровень обслуживания. В программном обеспечении Cisco IOS версии 12.0(7), 12.0 количество основного релиза, и 7 его уровень техобслуживания. Завершенный номер релиза 12.0 (7). Аналогичным образом, 12.1 – основная версия, а 12.1(3) – третья отладочная Cisco IOS Software Release 12.1. основной версии 12.1.

Версии ограниченного развертывания (LD)

LD является фазой зрелости Cisco IOS между FCS и общим развертыванием для основных релизов. Релизы для первоначального развертывания Cisco IOS только живут в фазе ограниченного развертывания, потому что они никогда не достигают сертификации GD.

Версии общего развертывания (GD)

В некоторый момент во время жизненного цикла релиза, Cisco объявит, что основной релиз готов к сертификации GD. Только главная версия может получить статус GD. Рекомендации по сертификации GD считаются удовлетворенными в следующих случаях:

- Доказано через дорогостоящее представление рынку в разнообразных сетях.
- Уточнение по метрикам, анализированным на стабильность и на возможные ошибки.
- Проверено исследованиями уровня удовлетворенности клиентов.
- Сокращение нормализованной тенденции клиента нашло дефекты в выпуске по предыдущим четырем отладочным релизам.

Многопрофильная команда сертификации GD отдела защиты интересов заказчиков сочинила инженеров TAC, инженеров Advanced Engineering Services (AES), Разработки Теста системы, и Разработка Cisco IOS сформирована для оценки каждой серьезной неисправности выпуска. Данная группа дает окончательное одобрение на сертификацию GD. После того, как выпуск получит статус GD, все его последующие версии также будут иметь статус GD. Следовательно, как только выпуск объявлен GD; это автоматически вводит фазу ограниченного обслуживания. На этом этапе изменение технологии кода, включая исправления ошибок со значительной переделкой кода, строго ограничено и контролируется диспетчером программ. Это гарантирует отсутствие ошибок в GD-сертифицированной версии программного обеспечения Cisco IOS. GD достигается конкретной обновленной версией. Обновления последующего обслуживания для того

выпуска являются также версиями GD. Например, программное обеспечение Cisco IOS версии 12.0 получило сертификацию GD в 12.0 (8). Таким образом Cisco IOS Software Release 12.0 (9), 12.0 (10), и так далее являются версиями GD.

Экспериментальный или диагностические образы

Когда важные проблемы программного обеспечения были определены, экспериментальный или диагностические образы иногда упоминаются как инженерные настройки и только созданы. Эти образы не являются частью обычного процесса релиза. Образы в этой категории являются сборками для конкретного заказчика, разработанными, чтобы помочь диагностировать проблему, тестировать исправление ошибки или предоставлять непосредственное исправление. Непосредственное исправление может быть предоставлено, когда это не опция для ожидания следующего промежуточного периода или отладочного релиза. Экспериментальный или диагностические образы может быть основан на любом ядре поддерживаемого программного обеспечения включая обслуживание или промежуточные версии любого типа релиза. Никакие официальные соглашения об именовании не существуют, но во многих случаях разработчик добавит начальные буквы, exp (для экспериментального), или дополнительные цифры к названию базового образа. Эти образы поддерживаются только временно, в связи с работой отдела разработок Cisco, поскольку операционные отделы выпусков Cisco TAC и Cisco IOS не сохраняют сопроводительные документы, например, таблицы символов или базовую историю образа. Эти образы не подвергаются никакой внутренней проверке Cisco.

Этапы жизненного цикла выпуска

В некоторый момент версии GD заменены более новыми версиями с последними сетевыми технологиями. Таким образом, процесс выпуска предыдущих версий был основан на таких трех принципах:

- **Конец продаж (EOS)** — Для основных релизов, дата EOS спустя три года после даты First Commercial Shipment (FCS). Это назначает последнюю дату, в течение которой выпуск может быть куплен для новых систем. Выпуск EOS и дальше будет доступен для загрузки на веб-сайте Cisco Connection Online (CCO) для обновления.
- **Конец разработки (EOE)** — выпуск EOE является последним отладочным релизом для выпуска GD, и, как правило, неотступно следует спустя три месяца после выпуска EOS. Клиенты могут продолжить получать техническую поддержку от Центра технической поддержки Cisco, а также загружать выпуск EOE от CCO. Бюллетень по продуктам, в котором анонсируются выпуски и даты EOS и EOE, публикуется один раз в год перед запланированной датой EOS. В это время клиенты должны начать исследовать обновление их программного обеспечения Cisco IOS для использования преимуществ последних сетевых технологий.
- **Окончание срока службы (EOL)** — В конце жизненного цикла релиза, вся поддержка Cisco IOS Software Release завершена и больше не доступная для загрузки в дату EOL. В целом дата EOL спустя пять лет после даты EOE. Информационный листок продукта EOL публикован приблизительно один год до фактической даты EOL.

Соглашение о записи имен версии Cisco IOS

В соглашении о записи имен образов Cisco IOS содержится полный профиль всех выпущенных образов. Название всегда включает идентификатор основного релиза и

идентификатор отладочного релиза. Имя может также включать указатель серии, обозначение повторной сборки (для служебных версий), обозначения специфических для служебного подразделения (BU) функций и идентификаторы специфических функций повторной сборки. Формат может быть сломан следующим образом:

Раздел соглашения о записи имен	Пояснение
x. y	Комбинация двух отдельных (один или два) цифровые идентификаторы, разделенные a'.', который определяет Значение Основного релиза. Это значение определено маркетингом Cisco IOS. Пример: 12.1
z	Одна - три цифры, который определяет отладочный релиз x.y. Это происходит каждые восемь недель. Значения равны 0 для бета-версии, 1 для FCS и 2 для первой отладочной версии. Пример: 12.1 (2)
p	Одна буква, которая определяет восстанавливание x.y (z). Значение начинается с символа "a" в нижнем регистре для первого восстановления, потом "b" и так далее. Пример: 12.1 (2a)

O	Одна - три альфа-буквы являются указателем последовательности релизов и являются обязательными для CTED, STED и X версий. Это также определяет семейство продуктов или платформы. Выпуски технологии ED используют две буквы. Первая буква обозначает технологию, а вторая служит отличительным признаком. Пример: `A = Access Server/Dial technology (example:11.3AA)`
	`B = Broadband (example:12.2B)`
	`D = xDSL technology (example:12.2DA)`
	`E = Enterprise feature set (example:12.1E)`
	`H = SDH/SONET technology (example:11.3HA)`
	`N = Voice, Multimedia, Conference (example:11.3NA)`
	`M = Mobile (example:12.2MB)`
	`S = Service Provider (example:12.0S)`
	`T = Consolidated Technology (example:12.0T)`
	`W = ATM/LAN Switching/Layer 3 (example:12.0W5)`"X" в первой позиции названия релиза определяет разовый релиз на основе CTED "T" серия. Например, XA, XB, XC, и так далее. "X" или "Y" во второй позиции названия релиза определяет кратковременный релиз раннего внедрения (ED) на основе, или связанный с,

	Релиз STED. Например, 11.3NX (на основе 11.3NA), 11.3WX (на основе 11.3WA), и так далее.
o	Необязательное обозначение модифицированной сборки конкретного выпуска (одна или две цифры). Оставьте незаполненный, не представляя восстанавливание. Запускается с 1, тогда 2, и т.д. Пример: 12,1(2)T1, 12,1(2)XE2
u	Однозначный или двузначный числовой указатель, определяющий специфическую для версии BU функциональность. Значение определяется группой маркетинга BU. Пример: 11.3 (6) WA4, 12.0 (1) W5
v	Одно- или двухразрядный цифровой указатель, указывающий поддерживаемый выпуск особого BU кода. Значения для выпусков: 0 - бета-версии, 1 - версии первой коммерческой поставки, 2 - версии планового обслуживания. Пример: 11.3(6)WA4(9), 12.0(1)W5(6)
p	Буква, обозначающая модифицированную сборку конкретного технологического выпуска. Значение начинается со строчной буквы "a" для первого восстановления, затем с "b" и так далее. Пример: 11.3(6)WA4(9a) является новой сборкой версии 11.3(6)WA4(9).

На следующей схеме показаны различные разделы соглашения об именовании в Cisco IOS:

Приложение B - надежность Cisco IOS

Надежность Cisco IOS является областью, где Cisco непрерывно стремится улучшиться. Прежде, чем обсудить ориентированные на клиента оптимальные методы, некоторое понимание качества внутреннего ПО Cisco IOS и работ по повышению надежности необходимо. Эти разделы предназначены непосредственно для предоставления обзора более свежих результатов работы Cisco над качеством программного обеспечения Cisco IOS, а также того, что должен взять на себя клиент касательно надежности программного обеспечения.

Программа контроля качества Cisco IOS

Cisco имеет четко определенный процесс разработки IOS, названный Большой технической методологией (GEM) GEM. Этот процесс имеет трехфазный жизненный цикл:

- Стратегия и планирование
- Выполнение
- Развертывания

Общие области в течение жизненного цикла включают приоритизацию введения функции, разработку, процесс тестирования, фазы внедрения ПО, Первого поставленного клиента

(FCS), GD и разработку поддержки. Cisco также придерживается многих лучших методов качества программного обеспечения рекомендации от организаций, таких как Международная организация по стандартам (ISO), Telcordia (раньше Bellcore), IEEE и Институт программной инженерии Карнеги Меллона. Эти рекомендации включены в процессы GEM Cisco. Процессами разработки программного обеспечения Cisco является сертифицируемый ISO 9001 (1994).

Первичный процесс для улучшения качества программного обеспечения Cisco IOS – это процесс, запускаемый клиентом, благодаря которому Cisco прислушивается к клиентам, определяет цели и метрики, воплощает лучшие идеи и контролирует результаты. Перекрестная организационная команда, которая стремится улучшать качество программного обеспечения, ведет этот процесс. Схему процесса повышения качества Cisco IOS показывают ниже:

Процесс повышения качества имеет конкретные измеримые цели для FY2002 и вне. Основная задача этих целей заключается в уменьшении дефектов путем заблаговременного определения проблем программного обеспечения в цикле тестирования, уменьшении количества неустраненных дефектов, улучшении единообразия функций и понятности версий программного обеспечения, а также в снабжении программным обеспечением и согласованными предсказуемыми планировщиками версий. Инициативы обратиться к этим областям включают новые программные средства тестового покрытия (определяющий области более слабого тестового покрытия), тестируют усовершенствование процесса корректирующего действия и системные усовершенствования регрессионного тестирования Cisco IOS. Дополнительные ресурсы были применены для решения этих проблем и существуют исполнительные и межфункциональные обязательства по всем основным Cisco IOS Software Release.

Тестирование Cisco IOS Release

Составляющая часть действий по обеспечению качества надежности ПО в Cisco является качеством, областью и покрытием тестирования. В целом, Cisco имеет следующие задачи по обеспечению качества IOS:

- Устранение найденных дефектов регрессии Cisco Internal. Это включает более высокое качество в разработку и идентификацию больших проблем в статическом / динамическом анализе.
- Уменьшите найденные дефекты клиента
- Уменьшение общего количества крупных дефектов
- Ясность выпуска ПО увеличения и непротиворечивость функции
- Предоставьте функции и отладочным релизам со списками и качеством

Внутренняя проверка Cisco может считаться процессом, где другие дефекты определены на других этапах тестирования. Общая задача должна найти правильные виды дефектов в правильной лабораторной работе. Это важно по нескольким причинам. Первым и самым важным является то, что достаточной полноты теста может не существовать на последних стадиях испытания. Затраты на проверку также серьезно возрастают с каждым последующим этапом, поскольку ранние этапы можно автоматизировать, а на более поздних возрастает сложность процессов и требуется большая компетентность. На следующей схеме показано тестирование спектра для Cisco IOS.

Первая стадия – разработка программного обеспечения. Cisco имеет несколько усилий в этой области, чтобы помочь улучшать начальное качество программного обеспечения.

Группы разработки также выполняют просмотры кода или даже множественные просмотры кода, чтобы гарантировать, что другие разработчики утверждают код новой характеристики или изменения ПО.

Следующий этап — проверка модулей. Поблочное тестирование использует программные средства, которые исследуют взаимодействие ПО без использования лабораторной работы. DevTest являются лабораторными испытаниями, которые включают проверку функциональных возможностей и регрессионное тестирование. Проверка функциональных возможностей разработана для исследования функциональности данной функции. Содержит настройку, сброс настроек и проверку перестановки всех возможностей в соответствии с определением в спецификации возможностей. Регрессионное тестирование проведено устройством автоматизированного тестирования, разработанным для проверки свойств функциональных возможностей и поведения на непрерывной основе. Основное внимание при тестирование уделяется проверке маршрутизации, коммутации и функционирования устройств в сетях с различными топологиями с помощью эхо-запросов и создания ограниченного трафика. Регрессионное тестирование только сделано на ограниченной комбинации функций, платформ, версий программного обеспечения и топологий к предельному числу возможных перестановок, однако более чем 4000 сценариев проверки регрессии используются сегодня. Проверка интеграции разработана, чтобы подробно остановиться на возможностях тестирования в лаборатории для большего количества полного семейства продуктов и совместимости. Проверка интеграции также увеличивает покрытие кода тестирования путем расширения тестирующий для включения проверок совместимости, напряжения и проверок производительности, тестов системы и проверки отрицательных состояний (тестирующий непредвиденные события).

На следующем лабораторном этапе предусмотрено сквозное тестирование для обычного пользовательского окружения. Их показывают в схеме выше как финансовый лабораторный тест (FTL) и NSITE, Тестирование Сценария Клиента. FTL был создан для обеспечения тестирования на критически - важные финансовые круги. NSITE является группой, которая предоставляет более подробно тестирование на другие технологии Cisco IOS. Лаборатории NSITE и FTL специализируются на таких областях, как масштабируемость и проверка производительности, возможность модернизации, доступность и устойчивость, совместимость и эксплуатационная надежность. Удобство обслуживания фокусируется на объемных проблемах инициализации, управление событиями / корреляция и устранение проблем под загрузкой. Другие лабораторные работы существуют в Cisco для других вертикальных рынков, чтобы помочь тестировать эти области.

Конечная лаборатория, показанная на диаграмме выше, обозначается как лаборатория клиента. Клиентская проверка является расширением действий по обеспечению качества и рекомендуемый для сред высокой доступности гарантировать, что была полностью протестирована точная комбинация функций, конфигурации, платформ, модулей и топологии. Тестирование должно включать проверку масштабируемости и производительности сети в указанной топологии, тестирование конкретных приложений, негативное тестирование в указанной конфигурации, проверку взаимодействия с устройствами других производителей (не Cisco) и испытания на принудительный отказ.

Программное обеспечение MTBF

Одна из наиболее распространенных метрик общей надежности является Mean Time Between Failure (MTBF). MTBF для надежности ПО полезен из-за возможностей анализа, которые были разработаны для надежности аппаратного обеспечения с помощью MTBF. Надежность аппаратного обеспечения может быть более точно определена с помощью

некоторых существующих стандартов. Cisco использует метод числа деталей на основе стандартных данных MTBF от Telcordia Technologies. Программное обеспечение MTBF, однако, не имеет никакой соответствующей методологии анализа и должно полагаться на полевое измерение для анализа MTBF.

В течение прошлых трех лет Cisco выполнила полевые измерения надежности ПО для Cisco, внутренняя сеть IT и эта работа задокументированы в Cisco. Работа основана на программных сбоях устройств Cisco IOS, которые могут быть измерены с помощью сведений управляющих прерываний протокола SNMP и сведений о периоде работоспособного состояния. Исследование определяет надежность ПО с помощью статистической логарифмически нормальной модели распределения для определенных выпусков ПО. Среднее время восстановления (MTTR) сбоя программного обеспечения основывается на среднем перезапуске маршрутизатора и временах восстановления. Шестиминутное время восстановления используется для сред предприятия, и пятнадцать минут используется для больших интернет-провайдеров (интернет-провайдеры). Результат этого продолжающегося исследования состоит в том, что программное обеспечение обычно встречает прекрасную доступность девяток, когда освобождено, или после нескольких версий обслуживания, и еще выше в течение долгого времени, столь же измеряется с помощью программных сбоев как единственный источник простоя. Исследования определили потенциальные значения MTBF в виде диапазона от 5000 часов для программ раннего развертывания до 50000 часов для основного программного обеспечения для развертывания.

Наиболее частым опровержением этой работы является тот факт, что аварийные ситуации, вызванные программным обеспечением, не учитывают время простоя, вызванное проблемами с надежностью программного обеспечения. Если эта метрика используется в усилиях по повышению качества, она может помочь улучшать скорость программных сбоев, но может проигнорировать другие критические области надежности ПО. Это примечание остается большей частью без ответа вследствие трудности точного прогнозирования надежности программного обеспечения с использованием статистических методов. Статистики по качеству программного обеспечения Cisco пришли к заключению, что больший типовой набор точного восстановления данных был бы необходим для надежного предсказания программного обеспечения MTBF с помощью более широкого диапазона типов простоя. Кроме того, теоретический статистический анализ произошел бы сложный из-за переменных, таких как сложность сети, экспертные знания штата для решения связанных проблем программного обеспечения, организации сети, функции включили, и процессы управления программным обеспечением.

В это время, никакая отрасль работают, был завершен, чтобы более точно предсказать надежность ПО с полевыми измерениями из-за трудности точного сбора этого типа уязвимых данных. Кроме того, большая часть Дона клиентов? t хотят сбор сведений о доступности, выполняемый Cisco непосредственно от их сети из-за составляющего собственность характера сведений о доступности. Некоторые организации действительно, однако, собирают данные по надежности ПО, и Cisco поощряет организации собирать метрики на доступности в связи с к простоям программного обеспечения и выполнять анализ корневых причин на тех простоях. Организации с более высокой надежностью ПО используют такую профилактическую стратегию для повышения надежности ПО с помощью ряда практических занятий, которыми они могут управлять.

Факторы, оказывающие влияние на надежность ПО

В результате отзывов клиентов упреждающие исследования, выполненные группой

технологий Cisco IOS и анализом основных причин, выполненным командой Расширенных сервисов Cisco, некоторыми более новыми предположениями и оптимальными методами, были сформированы что справка для улучшения надежности ПО. Данные обязательства сосредоточены на функциях проверки, степени зрелости или возраста программного обеспечения, включенных возможностях и количестве развернутых версий программного обеспечения.

Ответственность за проверку

Первое новое предположение касается ответственности за проверку. Cisco всегда ответственна за тестирование/проверку новых характеристик и функциональности, чтобы гарантировать, что они работают в новых продуктах. Cisco также ответственна за регрессионное тестирование, чтобы гарантировать, что новые версии программного обеспечения обратно совместимы. Однако Cisco не может проверить каждую функцию, топологию и платформу против каждого возможного предупреждения, которое пользовательское окружение может пустить в ход (особенности дизайна, загрузка и профили трафика). Оптимальные методы высокой доступности для клиентов включают тестирование в свернутую лабораторную топологию, которая подражает рабочей сети с помощью определенных функций клиента, дизайна, сервисов и трафика приложения.

Сравнение надежности и степени готовности программного обеспечения

Надежность программного обеспечения, как правило, является свидетельством его завершенности. Программное обеспечение готово к эксплуатации при получении демонстрации (использование) и корректировки идентифицированных ошибок. Операции релиза Cisco перешли к архитектуре выпуска серии, чтобы гарантировать, что программное обеспечение назревает без добавляемых новых характеристик. Клиенты, которые требуют высокой доступности, ищут более зрелое программное обеспечение с функциями, в которых они нуждаются теперь. Компромисс тогда существует между зрелостью программного обеспечения, требований доступности, и бизнесом - драйверами для новых характеристик или функциональности. Много организаций имеют стандарты или руководство по допустимой завершенности. Некоторые принимают только пятую промежуточную версию группы версий. Для других это может быть девятая сертификация или сертификация GD. Организация, в конечном счете, должна определить приемлемые уровни риска в отношении зрелости процессов разработки и обслуживания ПО.

Надежность возможностей и стандартов в сравнении с их количеством

Надежность ПО является также фактором того, сколько из кода протестировано и осуществлено в производственной среде. Как количество других аппаратных платформ и увеличений модулей, также увеличивается объем кода, осуществленный, который обычно увеличивает воздействие ошибок ПО. То же относится к числу настроенных протоколов, а также разнообразию реализованных конфигураций, топологий и решений. Дизайн, конфигурация, протоколы и факторы модуля оборудования могут способствовать на сумму кода, который осуществлен и к повышенному риску или воздействию ошибок ПО.

Теперь для операций по выпуску программного обеспечения имеется специализированное ПО, которое в основном ограничивает код, доступный в какой-либо определенной области. Служебные подразделения рекомендовали дизайны и конфигурации, которые более тщательно протестированы в Cisco и более широко используются клиентами. Клиенты также начали принимать оптимальные методы для стандартизированной модульной топологии и стандартных конфигураций, чтобы понизить сумму непротестированного воздействия кода и улучшить полную надежность ПО. В некоторых сетях высокого уровня

доступности действуют строгие правила поддержки стандартной конфигурации, стандарты модульной топологии и средства контроля версий программного обеспечения, что снижает риск введения непроверенного кода.

Надежность и число внедренных версий

Другой фактор надежности ПО является совместимостью между версиями и чистым объемом кода, который осуществлен с несколькими версий. Как количество увеличений версий программного обеспечения, также увеличивается объем кода, осуществленный, который тогда увеличивает воздействие ошибок ПО. В случае выполнения дополнительного кода различными версиями программного обеспечения угроза надежности системы возрастает почти экспоненциально. Это теперь распознано, что организации действительно должны выполнить по крайней мере ряд версий в сети для покрытия определенной функции и требований к платформе. Использование более пятидесяти версий в однородном сетевом окружении, однако, обычно служит признаком программных проблем вследствие невозможности достоверно анализировать или проверить такое большое количество версий.

Для улучшения надежности ПО развитие Cisco выполняет регрессионное тестирование программного обеспечения, чтобы гарантировать, что другие версии программного обеспечения совместимы. Кроме того, программный код является более модульным, и основные модули, менее вероятно, будут изменяться значительно между версиями в течение долгого времени. Операции релиза Cisco также изменили сумму программного обеспечения, доступного клиентам как версии с известными неисправностями, или проблемы совместимости быстро удалены из CCO, поскольку найдены дефекты.

Дополнительные сведения

- Операционные системы Сетевых технологий Cisco (IOS)
- Cisco Systems – техническая поддержка и документация