

Cisco IOS Management pour la gestion de réseaux à haute disponibilité : Livre blanc sur les pratiques recommandées

Contenu

[Introduction](#)

[Aperçu des pratiques recommandées de Cisco IOS](#)

[Aperçu de processus de gestion de cycle de vie de logiciel](#)

[Planification - Établir l'infrastructure de gestion de Cisco IOS](#)

[Stratégie et outils pour la planification de Cisco IOS](#)

[Définitions de piste de version de logiciel](#)

[Cycle et définitions de mise à jour](#)

[Processus de certification](#)

[Conception - Sélection et validation des versions de Cisco IOS](#)

[Stratégie et outils pour le Cisco IOS sélection et validation](#)

[Gestion de candidat](#)

[Test et validation](#)

[Implémentation - Déploiement rapide et réussi de Cisco IOS](#)

[Stratégie et outils pour des déploiements de Cisco IOS](#)

[Processus pilote](#)

[Implémentation](#)

[Exécutions - Gérer l'implémentation facilement disponible de Cisco IOS](#)

[Stratégies et outils pour des exécutions de Cisco IOS](#)

[Contrôle de version de logiciel](#)

[Gestion proactive de Syslog](#)

[Gestion de problème](#)

[Standardisation de configuration](#)

[Disponibilité de Gestion](#)

[Annexe A - Aperçu des versions de Cisco IOS](#)

[Échéances de cycles de vie de la version](#)

[Version de Cisco IOS nommant la convention](#)

[Annexe B - Fiabilité de Cisco IOS](#)

[Programme de qualité de Cisco IOS](#)

[Test de version de Cisco IOS](#)

[Logiciel MTBF](#)

[Suppositions de fiabilité de logiciel](#)

[Informations connexes](#)

[Introduction](#)

Le logiciel fiable le déployant et de mise à jour de Cisco IOS® est une priorité dans l'environnement de réseau essentiel d'affaires d'aujourd'hui qui exige de Cisco et de la concentration sur le consommateur renouvelés de réaliser la Disponibilité directe. Tandis que Cisco doit se concentrer sur leur engagement à la qualité logicielle, la conception de réseaux et les comités de soutien doivent également se concentrer sur des pratiques recommandées pour la gestion de logiciel de Cisco IOS. Le but est une efficacité plus facilement disponible et de gestion de logiciel. Cette méthode est un partenariat combiné pour partager, apprendre, et implémenter des pratiques recommandées de gestion de logiciel.

Ce document fournit un cadre opérationnel efficace des pratiques de gestion de Cisco IOS pour l'entreprise et les clients prestataires de services qui aident à favoriser la fiabilité de logiciel améliorée, la complexité réduite de réseau, et la Disponibilité accrue de réseau. Ce cadre aide également à améliorer des efficacités de gestion de logiciel en identifiant des domaines de responsabilité et des superpositions dans le test et la validation de gestion de logiciel entre les exécutions de release de Cisco et la base de clients de Cisco.

[Aperçu des pratiques recommandées de Cisco IOS](#)

Les tableaux suivants fournissent un aperçu des pratiques recommandées de Cisco IOS. Ces tables peuvent être utilisées comme aperçu de Gestion des pratiques recommandées définies, une liste de contrôle d'analyse des écarts de passer en revue des pratiques de gestion en cours de Cisco IOS, ou comme cadre pour créer des processus autour de la Gestion de Cisco IOS.

Les tables définissent les quatre composants de cycle de vie de la Gestion de Cisco IOS. Débuts de chaque table avec une stratégie et outils récapitulatifs pour la zone identifiée de cycle de vie. Après la stratégie et les outils le résumé sont des pratiques recommandées spécifiques qui s'appliquent seulement à la zone définie de cycle de vie.

[Planification - Établissant l'infrastructure de gestion de Cisco IOS](#) — La planification est la phase initiale de Gestion de Cisco IOS requise pour aider une organisation à déterminer quand améliorer le logiciel, où améliorer, et quel processus sera utilisé pour tester et valider des images potentielles.

Pratique recommandée	Détail
<u>Stratégie et outils pour la planification de Cisco IOS</u>	Obtenir commencé par la gestion prévisionnelle de Cisco IOS commence par une estimation honnête des activités actuelles, le développement des buts réalisables, et la planification de projets.
<u>Définitions de piste de version de logiciel</u>	Identifie où la cohérence de logiciel peut être mise à jour. Une piste de logiciel peut être définie en tant que seul groupement de version de logiciel, différencié d'autres zones par seul zone géographique, Plateformes, module, ou exigences de fonctionnalité.
<u>Cycle et</u>	Des définitions de cycle de mise à jour peuvent

définitions de mise à jour	être définies en tant qu'étapes de base de qualité en logiciel et gestion du changement utilisés pour déterminer quand un cycle de mise à niveau de logiciel devrait être initié.
Processus de certification	Les étapes de processus de certification devraient inclure l'identification de piste, les définitions de cycle de mise à jour, la Gestion de candidat, le test/validation, et au moins une certaine utilisation pilote de production.

[Conception - Sélection et validation des versions IOS](#) — Avoir un procédé bien défini pour sélectionner et valider des versions de Cisco IOS aide une organisation pour réduire l'interruption imprévue due aux tentatives infructueuses de mise à jour et aux erreurs de logiciel non planifiées.

Pratique recommandée	Détail
Stratégie et outils pour le Cisco IOS sélection et validation	Définissez les procédés pour sélectionner, tester, et valider de nouvelles versions de Cisco IOS. Ceci inclut un laboratoire de test de réseau qui émule le réseau de production
Gestion de candidat	La Gestion de candidat est l'identification des conditions requises de version logicielle et des risques potentiels pour le matériel particulier et les ensembles de caractéristiques activés.
Test et validation	Le test et la validation est un aspect essentiel de gestion de logiciel et de Haute disponibilité de réseau. L'essai en laboratoire approprié peut de manière significative réduire le temps d'arrêt de production, aider à former le personnel de support réseau, et aider à rationaliser des procédés d'implémentation du réseau.

[Implémentation - Déploiement rapide et réussi de Cisco IOS](#) — Les processus d'implémentation bien définis permet une organisation déploient à rapidement et avec succès de nouvelles versions de Cisco IOS.

Pratique recommandée	Détail
Stratégie et outils	La stratégie de base pour des déploiements de Cisco IOS est d'exécuter la certification finale par l'intermédiaire d'un processus pilote et d'un

pour des déploiements de Cisco IOS	déploiement rapide utilisant des outils de mise à jour et un processus d'implémentation bien défini.
Processus pilote	Afin de réduire l'exposition potentielle et plus sans risque à la capture toutes questions restantes de production, un pilote de logiciel est recommandées. Le plan pilote individuel devrait considérer la sélection pilote, la durée pilote, et la mesure.
Implémentation	Après la fin de la phase pilote, la phase d'implémentation de Cisco IOS devrait commencer. La phase d'implémentation peut inclure plusieurs étapes pour assurer le succès et l'efficacité de mise à niveau de logiciel comprenant le lent-commencement, la certification finale, la préparation de mise à jour, l'automatisation de mise à jour, et la validation finale.

[Exécutions - Gérant l'implémentation facilement disponible de Cisco IOS](#) — Les pratiques recommandées pour des exécutions de Cisco IOS incluent le contrôle de version de logiciel, la Gestion de Syslog de Cisco IOS, la Gestion de problème, la standardisation de configuration, et la Disponibilité de Gestion.

Pratique recommandée	Détail
Stratégies et outils pour des exécutions de Cisco IOS	La première stratégie des exécutions de Cisco IOS est de maintenir l'environnement aussi simple comme possible, évitant la variation dans la configuration et les versions de Cisco IOS. La deuxième stratégie est la capacité d'identifier et résoudre rapidement des défauts de réseau.
Contrôle de version de logiciel	Le contrôle de version de logiciel est le processus de mettre en application seulement les versions de logiciel normalisées et de surveiller le réseau pour valider ou changer probablement le logiciel dû à la conformité de non-version.
Gestion proactive de Syslog	La collecte, la surveillance, et l'analyse de Syslog sont des processus de gestion de défaut recommandés pour résoudre plus de problèmes spécifiques de réseau de Cisco IOS il est

	impossible difficile ou à l'identifier que par des autres moyens.
Gestion de problème	Processus de gestion détaillés de problème qui définissent l'identification de problème, la collecte d'informations, et un chemin bien analysé de solution. Ces données peuvent être utilisées pour déterminer la cause principale.
Standardisation de configuration	Les standards de configuration représentent la pratique de créer et de mettre à jour les périphériques similaires standard de paramètres de configuration globale à travers et des services ayant pour résultat la cohérence au niveau de l'entreprise de configuration globale.
Disponibilité de Gestion	La Disponibilité de Gestion est le processus de l'amélioration de la qualité utilisant la Disponibilité de réseau comme mesure d'amélioration de la qualité.

[Aperçu de processus de gestion de cycle de vie de logiciel](#)

La Gestion de cycle de vie de logiciel de Cisco IOS est définie comme ensemble de planification, de conception, d'implémentation, et de processus opérationnels qui sont recommandés pour les réalisations de logiciel fiable et le réseau facilement disponible. Ceci inclut des processus pour sélectionner, valider, et mettre à jour des versions de Cisco IOS dans le réseau.

Le but de la Gestion de cycle de vie de logiciel de Cisco IOS est d'améliorer la Disponibilité de réseau en diminuant la probabilité des erreurs de logiciel identifiées par production ou des pannes de modification associées par logiciel/mise à jour. Les pratiques recommandées définies dans cette documentation ont été affichées pour réduire de tels défauts et changer des pannes basées sur l'expérience pratique de beaucoup de clients de Cisco et des Services avancés de Cisco team. La Gestion de cycle de vie de logiciel peut au commencement augmenter des dépenses, cependant, le coût de possession global inférieur peut être réalisé de moins pannes et mécanismes plus rationalisés de déploiement et de support.

[Planification - Établir l'infrastructure de gestion de Cisco IOS](#)

La planification est la phase initiale de Gestion de Cisco IOS requise pour aider une organisation à déterminer quand améliorer le logiciel, où améliorer, et quel processus sera utilisé pour tester et valider des images potentielles.

Les pratiques recommandées incluent les [définitions de piste de version de logiciel](#), le [cycle et les définitions de mise à jour](#), et la création d'un [processus de certification de logiciel interne](#).

[Stratégie et outils pour la planification de Cisco IOS](#)

Commencez la gestion prévisionnelle de Cisco IOS par une estimation honnête des activités actuelles, le développement des buts réalisables, et la planification de projets. L'auto-évaluation devrait être faite en comparant des pratiques recommandées dans ce document aux processus dans votre organisation. Les questions fondamentales devraient inclure ce qui suit :

- Mon organisation a-t-elle un processus de certification de logiciel que cela inclut le test/validation de logiciel ?
- Mon organisation a-t-elle des normes de logiciel de Cisco IOS avec une quantité limitée de versions de Cisco IOS s'exécutant dans le réseau ?
- Mon organisation a-t-elle la difficulté déterminant quand améliorer le logiciel de Cisco IOS ?
- Mon organisation a-t-elle la difficulté déployant le nouveau logiciel chacun des deux de Cisco IOS de manière efficiente et efficace ?
- Mon organisation a-t-elle des questions de stabilité de Cisco IOS après déploiement qui affectent sérieusement le coût du temps d'arrêt ?

Après l'estimation, votre organisation devrait commencer définissant des buts pour la gestion de logiciel de Cisco IOS. Début en récupérant ensemble un groupe croix-fonctionnel de gestionnaires et/ou de piste des groupes de planification d'architecture, de l'ingénierie, de l'implémentation, et des exécutions pour aider à définir des buts de Cisco IOS et des projets d'amélioration du processus. Le but des téléconférences initiales devrait être de déterminer des objectifs, des rôles et des responsabilités globaux, d'assigner des actions à entreprendre, et définir des calendriers du projet initiaux. En outre, définissez les facteurs de succès capital et les mesures pour déterminer des avantages de gestion de logiciel. Les mesures potentielles incluent :

- Disponibilité (due aux problèmes logiciels)
- coût de mises à niveau de logiciel
- durée requise pour des mises à jour
- nombre de versions de logiciel s'exécutant dans la production
- succès/taux d'échec de modification de mise à niveau de logiciel

En plus de l'infrastructure de gestion globale de Cisco IOS prévoyant, quelques organismes définissent également les téléconférences de planification actuelles de logiciel pour se produire mensuel ou fois par trimestre. Le but de ces téléconférences est de passer en revue le déploiement de progiciels en cours et de commencer prévoyant tous les nouveaux logiciels nécessaires. La planification peut inclure revisiter ou modifier des procédés en cours de gestion de logiciel, ou définir simplement des rôles et des responsabilités des différentes phases de gestion de logiciel.

Les outils pendant la phase de planification consistent seulement en outils de gestion de l'inventaire logiciel. Le gestionnaire d'inventaire du Resource Manager Essentials des CiscoWorks 2000 (RME) est l'outil principal utilisé dans cette zone. [Le gestionnaire d'inventaire CiscoWorks2000 RME](#) simplifie considérablement la Gestion de version des Routeurs et des Commutateurs de Cisco par les outils de génération de rapports basés sur le WEB qui signalent et trient des périphériques de Cisco IOS basés sur la version de logiciel, la plate-forme de périphérique, la taille de la mémoire, et le nom du périphérique.

[Définitions de piste de version de logiciel](#)

La première pratique recommandée de planification de gestion de logiciel de Cisco IOS identifie où la cohérence de logiciel peut être mise à jour. Une piste de logiciel est définie en tant que seul groupement de version de logiciel, différencié d'autres zones par seul zone géographique, Plateformes, module, ou exigences de fonctionnalité. De façon optimale, un réseau devrait exécuter seulement une version de logiciel. Ceci diminue considérablement des coûts associés par gestion de logiciel et fournit un environnement cohérent et facilement géré. Cependant, la réalité est que la plupart des organismes doivent exécuter plusieurs versions dans le réseau en raison de la caractéristique, de la plate-forme, du transfert, et de la Disponibilité de questions dans des zones spécifiques. Dans de nombreux cas, la même version ne travaille pas aux Plateformes hétérogènes. Dans d'autres cas, l'organisation ne peut pas attendre une version pour prendre en

charge toutes leurs conditions requises. Le but est d'identifier les moins pistes de logiciel pour le réseau avec la considération pour tester/validation, certification, et conditions requises de mise à jour. Dans de nombreux cas, l'organisation peut avoir légèrement plus de pistes pour diminuer le test/validation, la certification, et les coûts de mise à jour en général.

Le premier fait de différenciation est support de plate-forme. Typiquement, les Commutateurs de RÉSEAU LOCAL, les Commutateurs BLÊMES, les principaux Routeurs, et les Routeurs chacun de périphérie ont les pistes distinctes de logiciel. D'autres pistes de logiciel peuvent être nécessaires pour des caractéristiques spécifiques ou des services, tels que le Data-Link Switching (DLSw), Qualité de service (QoS), ou Téléphonie sur IP, particulièrement si cette condition requise peut être localisée dans le réseau.

Des autres critères est fiabilité. Essai de beaucoup d'organismes pour exécuter le logiciel le plus fiable vers le noyau et le centre de traitement des données de réseau, tout en offrant une plus nouvelle fonctionnalité avancée, ou un support matériel, vers la périphérie. D'autre part, les caractéristiques d'évolutivité ou de bande passante sont souvent les plus nécessaires dans des environnements de noyau ou de centre de traitement des données. D'autres pistes peuvent être nécessaires pour les Plateformes spécifiques, telles que de plus grands sites de distribution qui ont une plate-forme différente de routeur WAN. Le tableau suivant est une définition de piste de logiciel d'exemple pour une grande organisation d'entreprise.

Pi st e	Zone	Plates- formes matérie lles	Caractéristiques	Versio n de Cisco IOS	État de certifi cation
1	Principale commutat ion de RÉSEAU LOCAL	6500	QoS	12.1E(A8)	Test
2	Commuta teur d'accès LAN	2924XL 2948XL	Protocole Unidirectional Link Detection (UDLD), Protocole Spanning Tree (STP)	12.0(5 .2)XU	3/1/01 certifi é
3	Distributio n/accès de RÉSEAU LOCAL	5500 6509	Superviseur 3	5.4(4)	7/1/01 certifi é
4	Module de route switch de commutat eur de distributio n (RSM)	RSM	Acheminement de Protocole OSPF (Open Shortest Path First)	12.0(1 1)	3/4/02 certifi é
5	Distributio n BLÊME de	7505 7507 7204	Relais de trames OSPF	12.0(1 1)	11/1/0 1 certifi

	headend	7206			é
6	Accès WAN	2600	Relais de trames OSPF	12.1(8)	6/1/01 certifié
7	Connectivité IBM	3600	Headend de Protocole SDLC (Synchronous Data Link Control)	11.3(8)T1	11/1/00 certifié

Les affectations de piste peuvent également changer au fil du temps. Dans de nombreux cas, les caractéristiques ou le support matériel peuvent intégrer dans plus se piquent des versions de logiciel permettant à différentes pistes pour migrer par la suite ensemble. Une fois que des définitions de piste ont été définies, l'organisation peut employer d'autres processus définis pour migrer vers la cohérence et la validation de nouvelles versions. Les définitions de piste sont également un effort actuel. N'importe quand une nouvelle caractéristique, service, matériel, ou la condition requise de module est identifiée, une nouvelle piste devrait être considérée.

Les organismes souhaitant initier un processus de piste devraient commencer par des conditions requises nouvellement définies de piste, ou dans certains cas, des projets de stabilisation pour des réseaux existants. Une organisation peut également avoir quelques vulgarisations identifiables avec les versions de logiciel existantes qui peuvent rendre la définition en cours de piste possible. Dans la plupart des cas, le transfert rapide aux versions identifiées n'est pas exigé si le client a la stabilité du réseau suffisante. L'architecture de réseau, ou la construction du groupe, possède normalement le procédé de définition de piste. Dans certains cas, une personne peut être responsable des définitions de piste. Dans d'autres cas, la piste de projet est responsable de développer des logiciels nécessaires et de nouvelles définitions de piste basés sur différents projets. C'est également une bonne idée de passer en revue des définitions de piste sur une base trimestrielle pour déterminer si de nouvelles pistes sont exigées, ou si les vieilles pistes exigent la fusion ou l'améliorent.

Des organismes qui identifient et mettent à jour des pistes de logiciel avec le contrôle de version strict ont été affichés pour avoir le succès le plus élevé avec un nombre décroissant de versions de logiciel dans le réseau de production. Ceci mène généralement à la stabilité améliorée de logiciel et à la fiabilité du réseau globale.

Cycle et définitions de mise à jour

Des définitions de cycle de mise à jour sont définies en tant qu'étapes de base de qualité en logiciel et gestion du changement utilisés pour déterminer quand un cycle de mise à niveau de logiciel devrait être initié. Les définitions de cycle de mise à jour permettent à une organisation pour prévoir correctement pour un cycle de mise à niveau de logiciel et pour allouer les ressources exigées. Sans définitions de cycle de mise à jour, une organisation connaît typiquement une augmentation des questions de fiabilité de logiciel dues aux exigences de fonctionnalité dans les versions stables en cours. Une autre exposition pourrait être l'organisation manquant l'occasion correctement de tester et valider une nouvelle version avant que l'utilisation de production soit exigée.

Un important aspect de cette pratique l'identifie quand et dans quelle mesure des processus de planification de logiciel devraient être initiés. C'est dû au fait qu'une principale cause des problèmes logiciels active une caractéristique, un service, ou une capacité matérielle dans la production sans diligence, ou à améliorer nouvelle au Cisco IOS une version sans des

considérations de gestion de logiciel. Un autre problème n'améliore pas. En ignorant les cycles et les conditions requises normaux de logiciel, beaucoup de clients font face à la tâche difficile de la mise à niveau logicielle par un certain nombre de différentes versions principales. La difficulté est due aux tailles d'image, le comportement par défaut change, des modifications de l'interprète de niveau commande (CLI), et le protocole change.

Cisco recommande un cycle bien défini de mise à jour, basé sur les pratiques recommandées comme définies en ce document, pour être initié toutes les fois que la nouvelle fonctionnalité principale, service, ou support matériel est exigée. Le degré de certification et de test/de validation devrait être analysé (basé sur le risque), pour déterminer les conditions requises précises de test/validation. L'évaluation des risques peut être faite par la situation géographique, l'emplacement logique (noyau, distribution, ou couches d'accès), ou le nombre estimé de personnes/de clients affectés. Si la fonctionnalité principale ou la capacité matérielle est contenue dans la version en cours, quelques processus profilés de cycle de mise à jour devraient également être initiés. Si la caractéristique est relativement mineure, considérez le risque et puis décidez quels processus devraient être initiés. En outre, le logiciel devrait être mis à jour pendant deux ans ou moins pour aider à s'assurer que votre organisation reste relativement en cours et que le processus de mise à niveau n'est pas trop encombrant.

Les clients devraient également considérer le fait qu'aucun correctif de bogue ne sera fait aux suites de logiciels qui ont passé la fin de l'état de la vie (EOL). Une certaine attention devrait également être accordée aux conditions requises d'affaires puisque beaucoup d'environnements peuvent tolérer, ou même l'accueil, plus d'ajouts de caractéristique avec peu ou pas de test/processus de validation et un certain temps d'arrêt en résultant. Les clients devraient également considérer les données plus nouvelles recueillies dans des exécutions de release de Cisco quand vu leurs conditions de test requises. Une analyse des bogues et des causes principales a prouvé que l'immense majorité de causes principales de bogue étaient le résultat des développeurs codant dans le secteur affecté de logiciel. Ceci signifie que si une organisation ajoute une fonction particulière ou un module à leur réseau dans une release existante, là est la probabilité d'éprouver une bogue liée à cette caractéristique ou module, mais une probabilité beaucoup inférieure que la nouvelle caractéristique, matériel, ou module affecteront d'autres zones. Ces données devraient permettre à des organismes pour diminuer des conditions de test requises, en ajoutant de nouvelles caractéristiques ou les modules qui sont pris en charge dans des releases existantes, en testant seulement le nouveau service ou caractéristique en même temps qu'autre ont activé des services. Les données devraient également être considérées quand la mise à niveau logicielle basée sur quelques bogues essentielles fondent dans le réseau.

Le tableau suivant affiche les conditions requises recommandées de mise à jour pour une organisation d'entreprise facilement disponible importante :

Déclencheur de gestion de logiciel	Condition requise de cycle de vie de logiciel
Nouveau service réseau. Par exemple, un nouveau circuit principal atmosphère ou un nouveau service VPN.	Validation complète de cycle de vie de logiciel comprenant le nouveau test de caractéristique (en même temps qu'autre services activés), le test réduit de topologie, ce qui-si évaluation de performances, et test de profil d'application.
La nouvelle capacité de réseau n'est pas prise	La validation complète de cycle de vie de logiciel comprenant le

en charge dans la version logicielle en cours. Les exemples incluent QoS et Commutation multiprotocole par étiquette (MPLS).	nouveau test de caractéristique, en même temps qu'autre a activé des services, le test réduit de topologie, ce qui-si évaluation de performances, et test de profil d'application.
Nouvelle fonctionnalité principale ou module de matériel qui existe dans la version en cours. Par exemple, ajoutant un nouveau module de GigE, support de Multidiffusion, ou DLSW.	Processus de gestion de candidat. Pleine validation possible basée sur des conditions requises de release. Test limité/validation possibles si la Gestion de candidat identifie la version en cours comme potentiellement acceptable.
Ajout mineur de caractéristique. Par exemple, un périphérique TACACS pour le contrôle d'accès.	Considérez la Gestion de candidat basée sur le risque de la caractéristique. Envisagez de tester ou piloter la nouvelle caractéristique basée sur le risque.
Logiciel dans la production pour deux années ou un examen trimestriel de logiciel.	Gestion de candidat et décisions économiques quant à la Gestion complète de cycle de vie d'identifier la release défendable en cours.

Mises à jour de secours

Dans certains cas, les organismes font face à la nécessité d'améliorer le logiciel dû aux bogues catastrophiques. Ceci peut mener aux problèmes si l'organisation n'a pas une méthodologie de mise à jour de secours. Les problèmes avec le logiciel peuvent s'étendre des mises à niveau de logiciel de non pris en charge, où le logiciel est mis à jour sans la Gestion de cycle de vie de logiciel, aux situations où les périphériques de réseau tombent en panne continuellement, mais l'organisation n'améliore pas puisque la certification/test sur la prochaine release de candidat n'a pas été terminée. Cisco recommande un processus de mise à niveau de secours pour ces situations où le test limité et les pilotes sont exécutés dans moins de zones critiques d'affaires du réseau.

Si les erreurs catastrophiques se produisent sans le contournement apparent et le problème est erreur de logiciel associée, Cisco recommande que Cisco les prennent en charge soit entièrement engagé pour isoler le défaut et pour déterminer si ou quand une difficulté est disponible. Quand la difficulté est disponible, Cisco recommande un cycle de mise à jour de secours pour déterminer rapidement si le problème peut être réparé avec le temps d'arrêt limité. Dans la plupart des cas, une organisation exécute une version prise en charge du code et la difficulté de problème est disponible dans une plus nouvelle version intermédiaire existante du logiciel.

Les organismes peuvent également se préparer aux mises à jour potentielles de secours. La préparation inclut le transfert aux releases prises en charge de Cisco IOS et l'identification /development des versions de remplacement de candidat dans le même Cisco IOS s'exercent comme version certifiée. Le logiciel pris en charge est important puisqu'il signifie que le

développement de Cisco ajoute toujours des correctifs de bogue à la suite de logiciels identifiée. En mettant à jour le logiciel pris en charge dans le réseau, l'organisation ramène le moment en raison de validation à la base du code plus familière et plus stable. Typiquement, un remplacement de candidat est une nouvelle image de logiciel intermédiaire dans le même Cisco IOS s'exercent sans des ajouts de caractéristique ou de support matériel. Une stratégie de remplacement de candidat est particulièrement importante si l'organisation a lieu pendant la phase de premier adopteur d'une suite de logiciels particulière.

Processus de certification

Un processus de certification aide à s'assurer que le logiciel validé est uniformément déployé dans l'environnement de production de l'organisation. Les étapes de processus de certification devraient inclure l'identification de piste, les définitions de cycle de mise à jour, la Gestion de candidat, le test/validation, et une certaine utilisation pilote de production. Un processus simple de certification, cependant, aide toujours à s'assurer que des versions de logiciel cohérentes sont déployées dans les pistes identifiées.

Commencez un processus de certification en identifiant des personnes à partir d'architecture, la construction/déploiement, et les exécutions pour dessiner et gérer le processus de certification. Le groupe devrait d'abord considérer des buts d'affaires et des capacités de ressource pour s'assurer que le processus de certification aura continué le succès. Ensuite, assignez les personnes ou la responsabilité globale de groupes des étapes principales dans le processus de certification comprenant la Gestion de piste, les définitions de mise à jour de cycle de vie, le test/validation, et les pilotes. Chacune de ces zones devrait être définie, approuvée, et être formellement communiquée dans l'organisation.

Incluez également les instructions pour la qualité ou l'approbation à chaque phase du processus de certification. Ceci s'appelle parfois un processus de porte de qualité parce que certains critères de qualité doivent être remplis avant que le processus puisse se déplacer à l'étape suivante. Ceci aide à s'assurer que le processus de certification est efficace et vaut les ressources assignées. Généralement quand des questions sont trouvées avec la qualité dans une zone, le processus repousse l'effort une étape.

Les candidats de logiciel peuvent ne pas répondre aux critères définis de certification en raison de la qualité logicielle ou du comportement inhabituel. Quand on trouve des questions qui affectent l'environnement, l'organisation devrait avoir un processus plus profilé pour certifier une version intermédiaire postérieure. Ceci aide à réduire les besoins en matière de ressources et est généralement efficace si l'organisation peut comprendre ce qui a changé et des quels défauts ont été résolus. Il n'est pas peu commun pour qu'une organisation rencontre un problème avec un premier candidat et de certifie une release intérimaire postérieure de Cisco IOS. Les organismes peuvent également faire une certification limitée ou fournir des mises en garde si quelques problèmes existent et peuvent améliorer à une release entièrement certifiée postérieure quand un nouvel intérim a été validé. L'organigramme ci-dessous est un processus de base de certification et inclut des portes de qualité (un examen suivant chaque bloc) :

Conception - Sélection et validation des versions de Cisco IOS

Avoir une méthodologie bien définie pour sélectionner et valider des versions de Cisco IOS aide une organisation pour réduire l'interruption imprévue due aux tentatives infructueuses de mise à jour et aux erreurs de logiciel non planifiées.

La phase de conception inclut la Gestion et le test/validation de candidat. La Gestion de candidat est le processus utilisé pour identifier des versions spécifiques pour les pistes définies de logiciel. Le test/validation est une partie du processus de certification et s'assure que la version de logiciel identifiée est réussie dans la piste exigée. Le test/validation devrait être fait dans un environnement de travaux pratiques avec une topologie et une configuration réduites qui ressemble étroitement à l'environnement de production.

Stratégie et outils pour le Cisco IOS sélection et validation

Chaque organisation devrait avoir un procédé pour sélectionner et valider des versions standard de Cisco IOS pour le réseau commençant par un processus pour sélectionner la version de Cisco IOS. Une équipe croix-fonctionnelle d'architecture, d'ingénierie, et d'exécutions devrait définir et documenter le processus de gestion de candidat. Une fois qu'approuvé, le processus devrait être retourné au groupe approprié de la livraison. On le recommande également qu'un modèle standard de Gestion de candidat soit créé qui peut être mis à jour avec les informations de candidat pendant qu'elles sont identifiées.

Non tous les organismes ont un environnement de travaux pratiques sophistiqué qui peut facilement imiter l'environnement de production. Quelques organismes ignorent l'essai en laboratoire en raison des dépenses et de la capacité de piloter une nouvelle version dans le réseau sans principal impact commercial. Cependant, des organismes facilement disponibles sont encouragés à construire un laboratoire qui imite le réseau de production et à développer un test/processus de validation pour assurer la test-couverture élevée pour de nouvelles versions de Cisco IOS. Une organisation devrait accorder environ six mois pour construire le laboratoire. Pendant ce temps, l'organisation devrait fonctionner pour créer les plans de test et les processus spécifiques de s'assurer que le laboratoire sera utilisé à son plein avantage. Pour le Cisco IOS, ceci signifie la création des plans d'essai spécifiques de Cisco IOS pour chaque piste de logiciel exigé. Ces processus sont de plus grands organismes étant donné que beaucoup de laboratoires disparaissent inutilisés pour des introductions de produit nouveau et de logiciel.

Les sections suivantes décrivent brièvement la Gestion et le test/outils de validation de candidat pour les utiliser pour le Cisco IOS sélection et validation.

Outils de gestion de candidat

Remarque: Pour utiliser la plupart des outils fournis ci-dessous, vous devez être un utilisateur enregistré et vous devez être ouvert une session.

- [Notes de mise à jour](#) — Fournit des informations concernant le matériel, le module, et la prise en charge de fonctionnalité d'une release. Des notes de mise à jour devraient être examinées pendant la Gestion de candidat pour s'assurer que tous les matériel requis et support logiciel existe dans la release potentielle, et comprendre n'importe quel transfert émet comprenant différentes conditions requises de comportement par défaut ou de mise à jour.

Test et outils de validation

Le test et les outils de validation sont utilisés pour tester et valider des solutions réseau comprenant le nouveaux matériel, logiciel, et applications.

- **Générateurs du trafic** — Générez les flux de trafic multiprotocole et les débits de paquets crus utilisés pour modeler le débit à travers n'importe quel lien particulier utilisant des protocoles spécifiques. Les utilisateurs peuvent spécifier la source, le MAC de destination, et les

numéros de prise, ces valeurs peuvent être incrémentées aux étapes spécifiées ou peuvent être configurées pour être statiques/fixes ou dans des incréments aléatoires. Les générateurs du trafic peuvent générer les paquets pour les protocoles suivants : IP Internet Network Packet Exchange (IPX) DECnet Apple Xerox Network System (XNS) Protocole ICMP (Internet Control Message Protocol) Protocole IGMP (Internet Group Management Protocol) Service réseau sans connexion (CLNS) Protocole UDP (User Datagram Protocol) Service de réseau intégré virtuel (VIGNES) Paquets de liaison de données Les outils sont fournis par des [transmissions d'Agilent](#) et de [Spirent](#) .

- **Compteur/capture/décodeur de paquet (renifleur)** — permet au client sélectivement pour capturer et décoder des paquets à tous les paquets et couches liaison de données. L'outil a la capacité pour permettre à l'utilisateur pour spécifier les filtres, qui permet capturer des données seulement spécifiées de protocole. Les filtres autres permettent à l'utilisateur pour spécifier capturer les paquets appartenant une adresse IP particulière, un numéro de port ou une adresse MAC. Les outils sont fournis par des [Technologies de renifleur](#) .
- **Simulateur/émulateur de réseau** — Permet au client pour remplir tables de routage des Routeurs spécifiques, basées sur les conditions requises de réseau de production. Prend en charge la génération des Routeurs de Protocole RIP (Routing Information Protocol), OSPF, de Protocole IS-IS (Intermediate System-to-Intermediate System), de Protocole IGRP (Interior Gateway Routing Protocol), d'Enhanced IGRP (EIGRP), et de Protocole BGP (Border Gateway Protocol) IP. Les outils sont fournis par des [transmissions de PacketStorm](#) et des [transmissions de Spirent](#).
- **Émulateurs de session** — Générez les flux de trafic multiprotocole de fenêtre glissante et soyez capable d'envoyer les flux de trafic multiprotocole à travers le réseau de test vers le périphérique récepteur. Le périphérique récepteur fait écho les paquets de retour vers la source. Le périphérique de source vérifie le nombre de paquets envoyés, reçu, hors des paquets d'ordre, et des paquets d'erreurs. L'outil fournit également la flexibilité de définir les paramètres de fenêtre dans le Protocole TCP (Transmission Control Protocol), de ce fait étroitement imitant les sessions du trafic de client/serveur dans le réseau des travaux pratiques. Les outils sont fournis par [Empirix](#) .
- **Émulateurs de réseau de large échelle** — Aidez en testant l'évolutivité de plus grands environnements. Ces outils peuvent créer et injecter facilement le trafic de type de contrôle dans une topologie de travaux pratiques afin d'imiter plus étroitement un environnement de production. Les capacités des voisins incluent des injecteurs d'artère, des voisins de protocole, et de la couche 2 protocole. Les outils sont fournis par des [transmissions d'Agilent](#) et de [Spirent](#) .
- **Simulateurs BLÊMES** — Idéal pour le trafic de l'application de test d'entreprise où la bande passante et le retard sont potentiellement une question. Ces outils permettent à des organismes pour tester localement une application avec le retard et la bande passante prévus pour voir comment l'application fonctionne au-dessus du WAN. Ces outils sont employés souvent pour le Développement d'applications et pour l'application profilant des test-types chez des organisations d'entreprise. Adtech, une division des [transmissions de Spirent](#) et [Shunra](#) fournissent les outils BLÊMES de simulation.

Gestion de candidat

La Gestion de candidat est le processus d'identifier des conditions requises de version logicielle et des risques potentiels pour le matériel particulier et les ensembles de caractéristiques activés. L'il est recommandé que une organisation passent quatre à huit heures recherchant correctement des

logiciels nécessaires, des notes de mise à jour, des erreurs de logiciel, et des risques potentiels avant de piloter une release. Les contours suivants la base pour la Gestion de candidat :

- Identifiez les candidats de logiciel par l'intermédiaire des outils du Cisco Connection Online (CCO).
- Maturité de logiciel d'évaluation des risques, nouvelle caractéristique, ou support de code.
- Identifiez et dépistez les erreurs de programmation, les questions, et les conditions requises connues tout au long du cycle de vie.
- Identifiez le comportement de configuration par défaut de l'image sélectionnée.
- Maintien soutiennent et les candidats d'actualisation pour le candidat potentiel change.
- Bug scrub.
- Support de Services avancés de Cisco.

Identifier des candidats de logiciel est devenu plus complexe avec le nombre croissant de productions et de suites de logiciels de Cisco. CCO a maintenant plusieurs outils comprenant le planificateur de mise à jour de Cisco IOS, l'outil de recherche de logiciel, la matrice de compatibilité de logiciel-matériel, et l'outil de mise à jour d'un produit qui peut aider des organismes à identifier les candidats potentiels de release. Ces outils peuvent être trouvés chez <http://www.cisco.com/cisco/software/navigator.html>.

Ensuite, analysez le risque du logiciel de candidat potentiel. C'est le processus de comprendre où le logiciel réside actuellement sur la curve de maturité et puis peser les conditions requises pour le déploiement avec le risque potentiel du candidat de release. Par exemple, si une organisation souhaite mettre le logiciel tôt du déploiement (ED) dans un environnement facilement disponible essentiel, le risque et les besoins en matière de ressources associés pour la certification réussie devraient être considérés. Une organisation devrait au moins ajouter des ressources en gestion de logiciel pour que les situations de plus gros risque assurent le succès. D'autre part, si une version générale du déploiement (GD) est disponible que réponde aux besoins d'une organisation, puis moins de ressources en gestion de logiciel sont nécessaires.

Quand des releases et les risques potentiels sont identifiés, exécutez un bug scrub pour déterminer si des bogues catastrophiques identifiées existent qui empêcheraient potentiellement la certification. Le watcher de la bogue de Cisco, le navigateur de bogue, et les agents de watcher de bogue peuvent aider à identifier des problèmes potentiels et devraient être utilisés tout au long du cycle de vie de logiciel pour identifier des questions de sécurité potentielle ou de défaut.

Un nouveau candidat de logiciel devrait également être passé en revue pour le comportement potentiel de configuration par défaut. Ceci peut être accompli en examinant les notes de mise à jour pour la nouvelle image logicielle et en passant en revue des différences de configuration avec l'image potentielle chargée sur les Plateformes indiquées. La Gestion de candidat peut également inclure l'identification de soutiennent des versions ou aller-aux versions si la version choisie ne répond pas à des critères de certification à un moment du processus. En observant des bogues liées aux caractéristiques pour une piste spécifiée, une organisation peut mettre à jour les candidats potentiels pour la certification.

Les Services avancés de Cisco sont également un excellent outil pour la Gestion de candidat. Ce groupe peut fournir le plus de compréhension dans le processus de développement et la Collaboration entre un grand nombre d'experts industriels en matière de beaucoup de différents environnements de marché vertical. Typiquement, les meilleurs bug scrub ou capacités de Gestion de candidat existent dans le support de Cisco, dû au niveau de l'expertise et de la visibilité dans des versions de logiciel de série s'exécutant à d'autres organismes.

Test et validation

Le test et la validation est un aspect essentiel des pratiques recommandées de Gestion et du réseau facilement disponible, en général. L'essai en laboratoire approprié peut de manière significative réduire le temps d'arrêt, les aides pour former le personnel de support réseau, et les aides de production en rationalisant des procédés d'implémentation du réseau. Pour être efficace cependant, l'organisation doit allouer les ressources nécessaires pour établir et mettre à jour l'environnement de travaux pratiques approprié, pour appliquer les ressources nécessaires pour réaliser les essais corrects, et pour utiliser une méthodologie de test recommandée qui inclut la collecte de mesure. Sans l'un de ces zones, un test et un processus de validation peuvent ne pas répondre aux attentes d'une organisation.

La plupart des organisations d'entreprise n'ont pas l'environnement de travaux pratiques recommandé de test. Pour cette raison, beaucoup d'organismes ont déployé des solutions inexactement, ont éprouvé des pannes de modification de réseau, ou rencontré les problèmes logiciels qui pourraient avoir été isolés dans un environnement de travaux pratiques. Dans quelques environnements, c'est acceptable, car le coût du temps d'arrêt ne compense pas le coût d'un environnement de travaux pratiques sophistiqué. Dans beaucoup d'organismes cependant, le temps d'arrêt ne peut pas être toléré. Ces organismes sont fortement invités pour développer les laboratoires de test recommandés, tester des types, et des méthodologies de test pour améliorer la qualité de réseau de production.

Le laboratoire et l'environnement de test

Le laboratoire devrait être une zone d'isolement avec assez d'espace pour des bureaux, des établis, équipement de test, et des modules ou des étagères de matériel. La plupart des grands organismes auront besoin entre quatre à dix étagères de matériel pour imiter l'environnement de production. Une certaine Sécurité physique est recommandée pour aider à mettre à jour un environnement de test tandis que les tests sont en cours. Ceci aide à empêcher un essai en laboratoire d'être dû perturbé à d'autres priorités de laboratoire comprenant l'emprunt de matériel, la formation, ou les répétitions d'implémentation. La Sécurité logique est également recommandée pour empêcher les artères factices d'écrire le réseau de production ou le trafic indésirable de quitter le laboratoire. Ceci peut être fait avec des filtres et des listes d'accès étendues de routage sur un routeur de passerelle de laboratoire. La Connectivité au réseau de production est utile pour des téléchargements logiciels et l'accès au réseau des travaux pratiques de l'environnement de production.

La topologie de travaux pratiques devrait pouvoir imiter l'environnement de production pour tous les plans de test spécifiques. Le matériel de reproduction, la topologie du réseau, et les configurations de caractéristique est recommandé. Naturellement, la reproduction de la topologie réelle est presque impossible, mais ce qui peut être faite est de reproduire la hiérarchie et l'interaction de réseau entre les périphériques de production. C'est important pour l'interaction de protocole ou de caractéristique entre de plusieurs périphériques. Quelques topologies de test seront différentes basées sur les conditions requises de test de logiciel. Le Cisco IOS de périphérie WAN testant, par exemple, ne devrait pas exiger des périphériques ou le test de type de RÉSEAU LOCAL et peut seulement exiger des Routeurs de périphérie WAN et des routeurs de distribution BLÈMES. La clé est d'imiter la fonctionnalité de logiciel sans production de reproduction. Dans certains cas, des outils peuvent même être utilisés pour imiter le comportement de grande puissance tel que des tables de décomptes voisins et de routage de protocole.

Des outils sont nécessaires également pour aider avec quelques types de test en améliorant la capacité d'imiter l'environnement de production et de collecter des essais. Outils qui aident la

production imitatrice à inclure des collecteurs du trafic, des générateurs du trafic, et des périphériques BLÊMES de simulation. Smartbits est un bon exemple d'un périphérique qui peut collecter et rejouer le trafic réseau ou générer de grands volumes du trafic. Une organisation peut également tirer bénéfice des périphériques qui peuvent aider à collecter des données, telles que des analyseurs de protocole.

Le laboratoire exige également une certaine Gestion. Beaucoup de plus grands organismes ont un gestionnaire à plein temps de laboratoire qui a la responsabilité de gérer le réseau des travaux pratiques. D'autres organismes utilisent l'architecture existante et les équipes techniques pour la validation de laboratoire. Les responsabilités de Gestion de laboratoire incluent l'équipement d'atelier de commande et la recherche de valeur, câblage, Gestion d'espace physique, définissant des règles de laboratoire et la direction, établissement du programme de laboratoire, documentation de laboratoire, installant des topologies de travaux pratiques, écrivant des plans de test, exécutant des essais en laboratoire, et gérant les questions identifiées par potentiel.

Types de test

De façon générale, il y a beaucoup de différents types de test qui peuvent être faits. Avant qu'établissant un laboratoire et un plan de test de test complets qui peuvent tester tout dans une multitude de configurations, une organisation devrait comprendre les différents types de test, l'intention du test, et si la construction de Cisco, la vente technique, ou la représentation des clients devraient ou pourrait être responsable de certains des divers tests. Les plans de test de client couvrent généralement les types plus exposés de test. Le tableau suivant aide en comprenant les différents types de test, quand les essais devraient être réalisés, et les interlocuteurs responsables.

Des tests ci-dessous, le test approprié du positionnement de la caractéristique spécifique d'une organisation, la topologie, et l'ensemble d'applications est normalement la plupart d'objet de valeur. Il est important de savoir que Cisco réalise l'essai complet et de régression, toutefois Cisco ne peut pas tester le profil de l'application de votre organisation avec votre combinaison spécifique de topologie, de matériel, et de caractéristiques configurées. En fait, il est infaisable pour tester la gamme complète de caractéristiques, de matériel, de modules, et de permutations de topologie. Supplémentaire, Cisco ne peut pas tester l'Interopérabilité avec le tiers matériel. Cisco recommande que les organismes testent la combinaison précise du matériel, des modules, des caractéristiques, et de la topologie trouvée dans leur environnement. Ce test devrait être conduit dans un laboratoire, avec une topologie réduite représentant l'environnement de production de votre organisation avec d'autres test-types les prenant en charge tels que la représentation, l'Interopérabilité, la panne, et la brûlure.

Test	Aperçu de test	Responsabilité de test
Fonctionnalité et caractéristiques	Détermine si des modules les caractéristiques de Cisco IOS et de matériel de base de Cisco fonctionnent comme annoncé. Des options de fonctionnalité de caractéristique	Test de périphérique de Cisco

	<p>ou de module aussi bien que de configuration de caractéristique devraient être testées. La suppression et l'ajout de configuration devraient être testés. Le test de base de panne et le test de brûlure est inclus.</p>	
Régression	<p>Détermine si la caractéristique ou le module fonctionne en même temps que d'autres modules et caractéristiques, et si la version de Cisco IOS fonctionne en même temps que d'autres versions de Cisco IOS par rapport aux caractéristiques définies. Inclut de la brûlure et du test de panne.</p>	Test de régression de Cisco
Représentation de base de périphérique	<p>Détermine la représentation de base de la caractéristique ou du module pour déterminer si le Cisco IOS comporte ou des modules de matériel répondent à des exigences minimum sous le chargement.</p>	Test de périphérique de Cisco
Combinaison de	Détermine si les	Cisco teste des

<p>topologie/caractéristique/matériel</p>	<p>caractéristiques et les modules fonctionnent comme prévu dans une combinaison spécifique de topologie et de module/caractéristique/matériel. Ce test devrait inclure la vérification de protocole, la vérification de caractéristique, la vérification de commande show, le test de brûlure et le test de panne.</p>	<p>topologies annoncées par norme dans les laboratoires tels que des solutions d'entreprise machinant (ESE) et l'ingénieur de contrôle en réseau d'intégration de solutions (NSITE). Les clients facilement disponibles devraient tester des combinaisons de caractéristique/module/topologie au besoin, particulièrement avec le logiciel de premier adopteur et les topologies non standard.</p>
<p>Panne (Ce qui-si)</p>	<p>Inclut les types ou les comportements communs de panne qui peuvent se produire dans un environnement de caractéristique spécifique/module/topologie et l'incidence de fonctionnalité de potentiel. Le test de panne inclut l'échange de carte, les lien-instabilités, les défaillances de périphérique, les pannes de lien, et les échecs de carte.</p>	<p>Cisco est responsable du test de base de panne. Les clients sont finalement responsables des problèmes de performance de panne liés à l'évolutivité de leur environnement individuel. Le test de panne devrait être fait, si possible, dans l'environnement de travaux pratiques de client.</p>
<p>NetworkPerformance (Ce qui-si)</p>	<p>Étudie le chargement de périphérique par</p>	<p>Les clients sont finalement responsables du</p>

	<p>rapport à une caractéristique spécifique/à matériel/à combinaison de topologie. Le foyer est sur la capacité des périphériques et la représentation telle que la CPU, la mémoire, l'utilisation de la mémoire tampon, et l'utilisation de lien par rapport à un type de trafic de positionnement et les besoins en matière de ressources pour des protocoles, des voisins, le nombre d'artères, et d'autres caractéristiques. Le test aide à assurer l'évolutivité dans de plus grands environnements .</p>	<p>chargement et de l'évolutivité de périphérique. Des inquiétudes de chargement et d'évolutivité sont souvent soulevées par des ventes ou des Services avancés de Cisco et sont souvent testées avec des TP Cisco tels que les laboratoires de Preuve-de-concept de client (CPOC).</p>
<p>Correctif de bogue</p>	<p>S'assure que les correctifs de bogue réparent le défaut identifié.</p>	<p>Cisco teste des correctifs de bogue pour s'assurer que la bogue est réparée. Les clients devraient également tester pour s'assurer que la bogue qu'ils ont éprouvée est réparée et que la bogue ne casse aucun autre aspect du module ou de la caractéristique. Les releases de maintenance sont</p>

		régression testée mais les versions intermédiaires ne sont habituellement pas.
Gestion de réseau CSNA	Étudie des capacités de Gestion de Protocole SNMP (Simple Network Management Protocol), la précision de variable MIB SNMP, le support de déROUTement, et le support de Syslog.	Cisco est responsable de tester les caractéristiques SNMP, la fonctionnalité, et la précision de base de variable MIB. Les clients devraient valider des résultats de Gestion de réseau et sont finalement responsables de la stratégie et de la méthodologie de Gestion pour des déploiements de nouvelle technologie.
Émulation de réseau à grande échelle	L'émulation de réseau à grande échelle utilise des outils tels que la suite de simulateur du routeur d'Agilent et d'outil de test de Spirent pour simuler de plus grands environnements . Ceci peut inclure des voisins de protocole, des comptes du circuit virtuel permanent en relais de trame (PVC), des tailles de tables de routage, le cache entries, et d'autres ressources typiquement exigées dans la	Les clients de Cisco sont généralement responsables des aspects du test de simulation du réseau qui reproduit leur environnement de réseau, qui peut inclure le nombre de voisins de protocole de routage/contiguïtés et tailles associées de table de routage et d'autres ressources qui sont dans la production.

	production qui ne sont pas dans le laboratoire par défaut.	
Interopérabilité	Teste tous les aspects concernant la Connectivité au matériel de réseau tiers, particulièrement si l'Interopérabilité de protocole ou de signalisation est exigée.	Les clients de Cisco sont généralement responsables de tous les aspects du test d'Interopérabilité.
Brûlure	Étudie des ressources en routeur au fil du temps. Les tests de brûlure exigent typiquement d'un périphérique d'être sous un certain chargement avec la recherche sur l'utilisation de ressource comprenant la mémoire, la CPU, et les mémoires tampons au fil du temps.	Cisco réalise l'essai de base de brûlure. Le test de client est recommandé par rapport à de seules combinaisons de topologie, de périphérique et de caractéristique.

Méthodologie de test

Une fois qu'une organisation sait ce qu'elles testent, une méthodologie devrait être développée pour le processus de test. Le but d'une méthodologie de test de pratique recommandée est aider à s'assurer que le test convenu est complet, bien documenté, facilement reproductible, et valeur en termes de trouver des problèmes de production potentielle. La documentation et les scénarios de laboratoire de récréation est particulièrement important pour les versions ultérieures de test ou pour les correctifs de bogue de test fondez dans l'environnement de travaux pratiques. Les étapes d'une méthodologie de test sont affichées ci-dessous. Quelques étapes de test peuvent également être exécutées simultanément.

1. Créez une topologie de test qui simule l'environnement de production au test. Un

environnement de test de périphérie WAN peut inclure quelques principaux Routeurs et un routeur de périphérie seulement, alors qu'un test de RÉSEAU LOCAL peut inclure plus de périphériques qui peuvent mieux représenter l'environnement.

2. Configurez les caractéristiques qui simulent l'environnement de production. La configuration des périphériques de laboratoire devrait étroitement apparier le matériel et les configurations du logiciel prévus de périphérique de production.
3. Écrivez un plan de test, définissant des tests et des buts, documentant la topologie, et définissant les tests fonctionnels. Les tests incluent la validation de protocole de base, la validation de commande show, le test de panne, et le test de brûlure. Un exemple d'un test spécifique dans un plan de test est trouvé dans le tableau suivant.
4. Validez la fonctionnalité de routage et de protocole. Document ou résultats de **commande show** prévus par spécification de base. Les protocoles devraient inclure chacun des deux des protocoles de la couche 2 tels que des protocoles atmosphère, de Relais de trames, de Protocole CDP (Cisco Discovery Protocol), d'Ethernets et de spanning-tree aussi bien que de couche 3 tels que l'IP, l'IPX et la Multidiffusion.
5. Validez la fonctionnalité de caractéristique. Document ou résultats de **commande show** prévus par spécification de base. Les caractéristiques peuvent inclure des commandes de configuration globale et toutes les caractéristiques essentielles telles que l'Authentification, autorisation et comptabilité (AAA).
6. Simulez le chargement, qui serait prévu dans l'environnement de production. La simulation de chargement peut être faite avec des collecteurs/générateurs du trafic. Validez attendu des variables d'utilisation de périphérique de réseau comprenant la CPU, la mémoire, l'utilisation de la mémoire tampon, et la statistique d'interface avec une recherche sur n'importe quelle perte de paquets. Document ou résultats de **commande show** prévus par spécification de base.
7. Réalisez l'essai de panne où on s'attendrait à ce que le périphérique et le logiciel traitent ou empêchent sous le chargement. Par exemple, cardez la suppression, joignez le lien instable, le lien instable d'artère, et les saturations de diffusion. Assurez-vous que les dérouterments corrects SNMP sont générés basés sur les caractéristiques étant utilisées dans le réseau.
8. Les résultats de test de document et les mesures de périphérique comme tests devraient être reproductibles.

Nom de test	Basculement de Protocole HSRP (Hot Standby Router Protocol)
Configurations requises de test	Appliquez le chargement à l'interface de passerelle principale. Le trafic devrait rudement être 20% vers la passerelle du point de vue de station utilisateur et 60% entrant vers le point de vue de station utilisateur. En outre, augmentez le trafic à un chargement plus élevé.
Étapes de test	Surveillez STP et HSRP par l'intermédiaire des commandes show . Échouent la connexion d'interface de passerelle principale et puis récupèrent la connexion après que les informations soient collectées.
Mesures prévues	CPU pendant le Basculement. Affichez l'interface avant, pendant, et ensuite pour la passerelle primaire et secondaire. Show hsrp avant, pendant, et ensuite.

Résultats prévus	La passerelle principale bascule à l'autre passerelle de routeur dans deux secondes. les commandes show reflètent correctement la modification. Le Basculement à la passerelle principale se produit quand la Connectivité est restaurée.
Résultats réels	
Passez ou échouez	
Modifications requises pour réaliser le passage	

Mesures de périphérique

Pendant la phase de test, effectuez et documentez les mesures suivantes pour s'assurer que le périphérique exécute correctement :

- Utilisation de mémoire
- Chargements CPU
- Utilisation de mémoire tampon
- Statistique d'interface
- Tables de routage
- Élimination des imperfections spécifique

Les informations pour des mesures varient selon le test particulier étant mis en application. Il peut également y avoir les informations complémentaires pour la mesure selon les problématiques spécifiques étant adressées.

Pour chaque application qui est testée, les paramètres de mesure à assurer là n'est aucune incidence des performances défavorable sur l'application donnée. Ceci est terminé en utilisant une spécification de base de représentation qui peut être utilisée pour comparer la représentation pré et le déploiement de courrier. Les exemples pour des tests de mesure d'application incluent :

- La durée moyenne où il prend pour se connecter un réseau.
- La durée moyenne il porte à la copie de Systèmes de fichiers en réseau (NFS) par ensemble de fichiers.
- La durée moyenne où il prend pour lancer une application et pour l'obtenir incité avec le premier écran.
- D'autres paramètres spécifiques à l'application.

[Implémentation - Déploiement rapide et réussi de Cisco IOS](#)

Un processus d'implémentation bien défini permet à une organisation pour déployer efficacement de nouvelles versions de Cisco IOS.

La phase d'implémentation inclut le processus pilote et le processus d'implémentation. Le processus pilote s'assure que la version de Cisco IOS sera réussie dans l'environnement et le processus d'implémentation permet des déploiements rapides et réussis de Cisco IOS à grande échelle.

Stratégie et outils pour des déploiements de Cisco IOS

La stratégie pour des déploiements de Cisco IOS est d'exécuter la certification finale par l'intermédiaire d'un processus pilote et d'un déploiement rapide utilisant des outils de mise à jour et un processus d'implémentation bien défini.

Avant d'initier un processus de pilote de réseau, beaucoup d'organismes établissent les instructions pilotes générales. Les instructions pilotes devraient inclure des attentes pour tous les pilotes tels que des critères de succès, des emplacements pilotes acceptables, la documentation pilote, des attentes pilotes de propriétaire, des conditions requises de notification d'utilisateur, et des durées pilotes prévues. Une équipe croix-fonctionnelle de l'ingénierie, de l'implémentation, et des exécutions est normalement impliquée en établissant des instructions pilotes globales et un processus pilote. Une fois que le processus pilote a été créé, les différents groupes d'implémentation peuvent normalement conduire les pilotes réussis utilisant les méthodes identifiées de pratique recommandée.

Une fois qu'une nouvelle version de logiciel a été approuvée pour la certification de déploiement et de finale, les besoins d'organisation de commencer prévoir la mise à jour de Cisco IOS. Débuts de planification avec l'identification de nouvelles conditions requises d'image comprenant la plate-forme, la mémoire, l'éclair, et la configuration. L'architecture et les groupes de construction définissent normalement de nouvelles conditions requises d'image logicielle pendant la phase de Gestion de candidat du cycle de vie de Gestion de Cisco IOS. Une fois que les conditions requises ont été identifiées, chaque périphérique doit être validé, et être probablement mis à jour, par le groupe d'implémentation. Le module de gestionnaire de l'image logicielle CiscoWorks2000 (BAIN) peut également exécuter l'étape de validation en validant des conditions requises de Cisco IOS contre l'inventaire des périphériques. Quand tous les périphériques ont été validés et ou mis à jour aux nouvelles normes correctes d'image, le groupe d'implémentation peut commencer un processus d'implémentation de lent-commencement utilisant le module du BAIN CiscoWorks2000 comme outil de déploiement de progiciels.

Une fois que la nouvelle image a été avec succès déployée un certain nombre de fois, l'organisation peut commencer un déploiement rapide utilisant le BAIN de CiscoWorks.

Gestion des stocks de Cisco IOS

Le gestionnaire d'inventaire du Resource Manager Essentials CiscoWorks2000 (RME) simplifie considérablement la Gestion de version des Routeurs et des Commutateurs de Cisco par les outils de génération de rapports basés sur le WEB qui signalent et trient des périphériques de Cisco IOS basés sur la version de logiciel, la plate-forme de périphérique, et le nom du périphérique.

BAIN de Cisco IOS

Le BAIN CiscoWorks2000 peut aider à réduire les complexités sujettes aux erreurs du processus

de mise à niveau. Les liens intégrés à CCO corrént la documentation en ligne de Cisco au sujet des correctifs logiciels avec le logiciel de Cisco IOS et de Catalyst déployé dans le réseau, mettant en valeur les notes relatives en tech. Les nouveaux outils de planification trouvent des configurations système requises et envoient des notifications quand les mises à niveau matérielles (ROM de démarrage, RAM d'instantané) sont nécessaires pour prendre en charge les mises à jour proposées d'image logicielle.

Avant qu'une mise à jour soit initiée, les conditions préalables pour une nouvelle image sont validées contre les données d'inventaire de commutateur ou de routeur de cible pour aider à assurer une mise à jour réussie. Quand les plusieurs périphériques sont mis à jour, le BAIN synchronise des tâches de téléchargement et permet à l'utilisateur pour surveiller la progression du travail. Les travaux planifiés sont commandés par un processus d'approbation, permettant à des gestionnaires d'autoriser les activités d'un technicien avant d'initier chaque tâche de mise à jour. RME 3.3 inclut la capacité d'analyser des mises à niveau de logiciel pour des Plateformes de Cisco IGX, BPX, et MGX, considérablement simplifiant et réduisant la durée requise pour déterminer l'incidence d'une mise à niveau de logiciel.

Processus pilote

Afin de réduire l'exposition potentielle et plus sans risque à la capture toutes questions restantes de production, un pilote de logiciel est recommandées. Les pilotes sont généralement plus importants pour des déploiements de nouvelle technologie, toutefois beaucoup de nouveaux déploiements de progiciels seront liés à de nouvelles services, caractéristiques, ou matériel, où un pilote est plus essentiel. Le plan pilote individuel devrait considérer la sélection pilote, la durée pilote et la mesure. La sélection pilote est le processus de identifier quand et où un pilote devrait être fait. La mesure pilote est le processus de collecter les données exigées pour identifier le succès et échec ou les problèmes potentiels.

La sélection pilote identifie où et comment un pilote sera terminé. Un pilote peut commencer par un périphérique dans une zone de bas-incidence et étendre à de plusieurs périphériques dans une zone plus à haute impression. Quelques considérations pour la sélection pilote où l'incidence peut être réduite incluent ce qui suit :

- Installé dans une zone du réseau résilient sur une incidence à un dispositif due à la Redondance.
- Dans une zone du réseau avec un nombre minimal d'utilisateurs derrière le périphérique sélectionné qui peut traiter une certaine incidence possible de production.
- Consider séparant le pilote le long des lignes d'architecture. Par exemple, pilotez-le dans l'accès, la distribution, et/ou les principales couches du réseau.

La durée de ce pilote devrait être basée sur le temps où elle prend suffisamment pour tester et évaluer toutes les caractéristiques de périphérique. Ceci devrait inclure la brûlure et le réseau sous les charges de la circulation normales. La durée dépend également de l'étape dans la mise à niveau du code et de la zone du réseau où le Cisco IOS s'exécute. Si le Cisco IOS est une nouvelle version principale, une plus longue période pilote est préférée. Considérant que si la mise à jour est une release de maintenance avec de nouvelles configurations minimales, une période pilote plus courte suffira.

Pendant la phase pilote il est important de surveiller et documenter des résultats d'une manière semblable comme test initial. Ceci peut inclure les enquêtes d'utilisateur, la collecte des informations pilote, la collecte de problème, et les critères de succès/panne. Les personnes devraient être directement responsables du cheminement et la progression pilote de surveillance pour assurer toutes les questions sont identifiées et que des utilisateurs et les services impliqués

dans le pilote sont satisfaits avec les résultats pilotes. La plupart des organismes certifieront une release si elle est réussie dans un pilote ou un environnement de production. Cette étape est une panne essentielle dans quelques environnements dus à un succès perçu quand aucun critère de mesure ou de succès n'est identifié ou est documenté.

Implémentation

Après la phase pilote a été terminée dans le réseau de production, commencent la phase d'implémentation de Cisco IOS. La phase d'implémentation inclut plusieurs étapes pour assurer le succès de mise à niveau de logiciel et l'efficacité d'implémentation, y compris le début lent d'implémentation, la certification finale, la préparation de mise à jour, l'automatisation de mise à jour, et la validation finale.

Le lent-commencement d'implémentation est le processus de mettre en application lentement une release nouvellement testée pour s'assurer que l'image a la pleine exposition à l'environnement de production avant certification finale et conversion complète. Quelques organismes peuvent commencer par un périphérique et un jour d'exposition avant de passer au périphérique deux améliore le jour suivant et peut-être quelques plus le jour suivant. Quand approximativement dix périphériques ont été placés dans la production, l'organisation peut attendre jusqu'à une à deux semaines avant la certification finale de la version particulière de Cisco IOS. Sur la certification finale, l'organisation peut plus rapidement déployer la version identifiée avec un niveau beaucoup plus élevé de confiance.

Après que le procédé lent de début, tous les périphériques identifiés pour la mise à jour devrait être passé en revue et validé utilisant l'inventaire des périphériques et une matrice des normes minimum de Cisco IOS pour que le bootstrap, la mémoire vive dynamique, et l'éclair s'assure que les exigences sont répondues. Les données peuvent être saisies par les outils internes, des outils SNMP de tiers, ou par l'utilisation du CiscoWorks2000 RME. Le BAIN CiscoWorks2000 passe en revue ou examine ces variables avant l'implémentation. Cependant, c'est toujours une bonne idée de savoir quoi prévoir pendant l'implémentation tente.

Si plus de cent périphériques semblables sont programmés pour des mises à jour, on le recommande fortement qu'une méthode automatisée soit utilisée. L'automatisation a été affichée pour améliorer l'efficacité de mise à jour et pour améliorer le pourcentage des succès de mise à jour de périphérique pendant de grands déploiements, basé sur une mise à jour interne de 1000 périphériques avec et sans le BAIN. Cisco recommande que les CiscoWorks 2000 NAGENT soient utilisés pour de grands déploiements dus au degré de vérification qui est exécutée pendant la mise à jour. Le BAIN soutiendra même d'une version de Cisco IOS si un problème est détecté. Fonctions de BAIN en créant et en programmant les travaux de mise à jour, où un travail est configuré avec les périphériques, les images désirées de mise à jour, et le délai d'exécution du travail. Chaque travail devrait contenir douze ou moins de mises à jour de périphérique, et les jusqu'à douze travaux peuvent fonctionner simultanément. Le BAIN vérifie également que la version programmée de mise à jour de Cisco IOS exécute suivre avec succès la mise à jour. Il est recommandé pour accorder approximativement vingt minutes pour chaque mise à jour de périphérique (vérification y compris). Utilisant cette formule, une organisation peut améliorer trente-six périphériques par heure. Cisco recommande également qu'un maximum de cent périphériques soit mis à jour par soirée pour réduire l'exposition de problème potentiel.

Après une mise à jour automatisée, une certaine validation devrait être faite pour assurer le succès. L'outil du BAIN CiscoWorks2000 peut exécuter les scripts personnalisés suivant la mise à jour pour exécuter davantage de vérification de succès. La vérification inclut la validation que le routeur a le numéro approprié d'artères, s'assurant que logique/interfaces physiques soyez haut et

en activité, ou validant que le périphérique est accessible. La liste de contrôle suivante témoin peut entièrement valider le succès d'un déploiement de Cisco IOS :

- Le périphérique a-t-il correctement rechargé ?
- Le périphérique est-il pingable et accessible par l'intermédiaire des Plateformes du système d'administration de réseaux (NMS) ?
- Les interfaces prévues sont-elles sur le périphérique en hausse et l'active ?
- Le périphérique a-t-il les contiguïtés correctes de protocole de routage ?
- La table de routage est-elle remplie ?
- Le périphérique passe-t-il le trafic correctement ?

Exécutions - Gérer l'implémentation facilement disponible de Cisco IOS

Les exécutions facilement disponibles de pratique recommandée de l'environnement de Cisco IOS aide à réduire la complexité de réseau, à améliorer le temps de résolution des problèmes, et à améliorer la Disponibilité de réseau. La section d'exécutions de Gestion de Cisco IOS inclut la stratégie, les outils, et les méthodologies de pratique recommandée recommandées pour gérer le Cisco IOS.

Les pratiques recommandées pour des exécutions de Cisco IOS incluent le contrôle de version de logiciel, la Gestion de Syslog de Cisco IOS, la Gestion de problème, la standardisation de configuration, et la Disponibilité de Gestion. Le contrôle de version de logiciel est le processus de dépister, de valider, et d'améliorer la cohérence de logiciel dans les pistes identifiées de logiciel. La Gestion de Syslog de Cisco IOS est le processus proactivement de la surveillance et de l'action sur des messages plus prioritaires de Syslog générés par Cisco IOS. La Gestion de problème est la pratique de rapidement et efficacement collectant les informations essentielles de problème pour des questions connexes de logiciel afin d'aider à empêcher de futures occurrences. La standardisation de configuration est le processus de normaliser des configurations pour réduire le potentiel pour que le code non essayé soit exercé dans la production et pour normaliser le protocole réseau et le comportement de caractéristique. La Disponibilité de Gestion est le processus d'améliorer la Disponibilité basé sur des mesures, des buts d'amélioration, et des projets d'amélioration.

Stratégies et outils pour des exécutions de Cisco IOS

Beaucoup de stratégies et d'outils de qualité existent pour aider à gérer des environnements de Cisco IOS. La première stratégie principale pour des exécutions de Cisco IOS est de maintenir l'environnement aussi simple comme possible, évitant la variation dans la configuration et les versions de Cisco IOS autant que possible. La certification de Cisco IOS a été déjà discutée, toutefois la cohérence de configuration est une autre zone clé. L'architecture/le groupe de construction devrait être responsables de créer des standards de configuration. L'implémentation et le groupe d'exécutions ont alors la responsabilité de configurer et mettre à jour les normes par le Cisco IOS contrôle de version et les standards de configuration/contrôle.

La deuxième stratégie pour des exécutions de Cisco IOS est la capacité d'identifier et résoudre rapidement des défauts de réseau. Des problèmes de réseau devraient généralement être identifiés par le groupe d'exécutions avant que les utilisateurs les appellent dedans. Des problèmes devraient également être résolus aussi rapidement que possible sans davantage d'incidence ou de modification à l'environnement. Quelques uns les meilleures pratiques

principales dans cette zone sont Gestion de problème et Gestion de Syslog de Cisco IOS. Un outil à aider rapidement à diagnostiquer des crash de logiciel de Cisco IOS est l'Output Interpreter de Cisco.

La troisième stratégie est à amélioration cohérente. Le processus primaire est d'améliorer une Disponibilité basée sur qualité de programme d'amélioration. En exécutant l'analyse de la cause d'origine sur toutes les questions, y compris des questions connexes de Cisco IOS, une organisation peut améliorer la couverture de test, améliorer des temps de résolution des problèmes, et améliorer les processus qui éliminent ou réduisent l'incidence de panne. L'organisation peut également regarder des problèmes courants et établir des processus pour résoudre ces problèmes plus rapides.

Les outils pour des exécutions de Cisco IOS incluent la gestion des stocks pour que le contrôle de version de logiciel (CiscoWorks2000 RME), la Gestion de Syslog pour parvenir des messages de Syslog, et les gestionnaires de configuration de périphérique gèrent la cohérence de configuration de périphérique.

Gestion de Syslog

Les messages de Syslog sont des messages envoyés par le périphérique à un serveur de collecte. Ces messages peuvent être des erreurs (par exemple, un lien allant vers le bas), ou ils peuvent être informationnels, comme quand quelqu'un a été dedans de configurer un terminal sur un périphérique.

Les outils de gestion de Syslog se connectent et les messages de Syslog de piste reçus par des Routeurs et des Commutateurs. Quelques outils ont des filtres pour permettre la suppression des messages indésirables qui peuvent amoindrir les importants. Les outils de Syslog devraient également permettre l'enregistrement à créer ont basé sur les messages reçus. L'enregistrement peut être affiché par délai prévu, périphérique, type de message, ou priorité de message.

L'outil de Syslog le plus populaire pour la Gestion de Cisco IOS est gestionnaire de Syslog CiscoWorks2000 RME. D'autres outils sont disponibles incluant SL4NT, un programme de shareware de [Netal](#) et l privé d'OpenSystems.

Configuration Manager de périphérique de CiscoWorks

La Configuration Manager du périphérique CiscoWorks2000 met à jour des archives actives et fournit une méthode facile de mettre à jour des modifications de configuration à travers de plusieurs Routeurs et Commutateurs de Cisco. Le gestionnaire de configuration surveille le réseau pour des modifications de configuration, met à jour les archives quand une modification est détectée, et se connecte les informations de modification au service d'audit de modification. Une interface utilisateur basée sur le WEB te permet pour rechercher les archives pour des attributs spécifiques de configuration et pour comparer le contenu de deux fichiers de configuration pour l'identification facile des différences.

Output Interpreter de Cisco

L'Output Interpreter de Cisco est un outil utilisé en diagnostiquant des crash forcés par logiciel. L'outil peut aider à identifier des erreurs de logiciel sans appeler le centre d'assistance technique Cisco (TAC), ou il peut être utilisé en tant qu'informations primaires au TAC après un crash forcé par logiciel. Ces informations aideront généralement à accélérer une résolution au problème, au moins en termes de collecte de l'information requise.

Contrôle de version de logiciel

Le contrôle de version de logiciel est le processus de mettre en application seulement les versions de logiciel normalisées et de surveiller le réseau pour valider ou changer probablement le logiciel dû à la conformité de non-version. Généralement le contrôle de version de logiciel fait utilisant un processus de certification et un contrôle de normes. Beaucoup d'organismes éditent des normes de version sur un web server central. En outre, le personnel d'implémentation est formé pour passer en revue quelle version s'exécute et pour mettre à jour la version si ce n'est pas des normes conformes. Quelques organismes ont un processus de porte de qualité où la validation secondaire est terminée par des audits pour s'assurer que la norme est suivie pendant l'implémentation.

Lors du fonctionnement, il n'est pas rare de voir des versions non standard dans le réseau, particulièrement si le réseau et l'équipe d'exploitation sont grands. Ceci peut être dû au plus nouveau personnel non formé, aux commandes SIG-configurées de démarrage, ou aux réalisations décochées. C'est toujours une bonne idée de valider périodiquement des normes de version de logiciel utilisant des outils tels que des CiscoWorks 2000 RME qui peuvent trier tous les périphériques par version de Cisco IOS. Quand des non-normes sont identifiées, elles devraient être immédiatement signalées et un dossier d'incident ou un ticket de modification soit initié pour apporter la version à la norme identifiée.

Gestion proactive de Syslog

La collecte, la surveillance, et l'analyse de Syslog sont des processus de gestion de défaut recommandés pour résoudre plus de problèmes spécifiques de réseau de Cisco IOS il est impossible difficile ou à l'identifier que par des autres moyens. La collecte de Syslog, la surveillance, et l'aide d'analyse pour améliorer le temps de résolution des problèmes en identifiant et en résolvant beaucoup de défauts proactivement avant que des problèmes plus sérieux de réseau soient expérimentés, ou sont signalées par des utilisateurs. Le Syslog fournit également plus de méthode efficace de collecter une grande variété de problèmes une fois comparé à interrogation cohérente SNMP pour un grand nombre de variables MIB. La collecte, la surveillance, et l'analyse de Syslog est accomplie en utilisant la configuration Cisco IOS correcte, des outils de corrélation de Syslog, tels que la gestion d'événement CiscoWorks2000 RME, et/ou de Syslog. La gestion d'événement de Syslog est faite en analysant les données collectées de Syslog pour les messages essentiels identifiés et puis en expédiant une alerte ou le déroutement à un gestionnaire d'événement pour la notification et la résolution en temps réel.

La surveillance de Syslog exige du support ou des scripts d'outil NMS d'aider à analyser et rendre compte des données de Syslog. Ceci inclut la capacité pour trier des messages de Syslog à la date ou le délai prévu, le périphérique, le type de message de Syslog, ou la fréquence de message. Dans les réseaux plus vastes, des outils ou les scripts peuvent être mis en application pour analyser des données de Syslog et pour envoyer des alertes ou des notifications aux systèmes de gestion d'événement ou aux exécutions et au personnel d'ingénierie. Si des alertes pour une grande variété de données de Syslog ne sont pas utilisées, l'organisation devrait examiner des données plus prioritaires de Syslog au moins quotidiennes et créer des dossiers d'incident pour des problèmes potentiels. Afin de détecter proactivement les problèmes de réseau qui ne peuvent être vus par la surveillance normale, examen périodique et l'analyse des données historiques de Syslog devrait être exécutée pour détecter les situations qui peuvent ne pas indiquer un problème immédiat, mais peut fournir une indication d'un problème avant qu'il devienne affecter de service.

Gestion de problème

Beaucoup de clients éprouvent l'interruption liée supplémentaire à un manque de processus en Gestion de problème. Le temps d'arrêt supplémentaire peut se produire quand essayez d'administrateurs réseau pour résoudre le problème utilisant rapidement une combinaison des commandes ou des modifications de configuration service-les affectant plutôt que passant le temps sur l'identification de problème, la collecte d'informations, et un chemin bien-analysé de solution. Le comportement observé dans cette zone inclut les périphériques de rechargement, ou les tables effaçantes de Routage IP avant d'étudier un problème et sa cause principale. Dans certains cas, ceci se produit en raison des premiers buts de la résolution des problèmes de support de niveau. Le but dans toutes les questions connexes de logiciel devrait être de collecter rapidement les informations nécessaires requises pour l'analyse de la cause d'origine avant de restaurer la Connectivité ou le service.

Un processus de gestion de problème est recommandé dans de plus grands environnements. Ce processus devrait inclure un certain degré de descriptions du problème par défaut et de collections appropriées de **commande show** avant la transmission des problèmes à un deuxième niveau. Le premier niveau de support devrait ne jamais effacer des artères ou recharger des périphériques. De façon optimale, la première organisation de niveau devrait rapidement collecter des informations et faire suivre à un deuxième niveau. En passant juste quelques plus de minutes au commencement sur l'identification ou la description du problème de problème, une détection de cause principale est beaucoup plus probable, de ce fait permettant un contournement, une identification de laboratoire, et un enregistrement de bogue. Le deuxième support de niveau devrait être bien exprimé en vers dans les types d'informations des lesquels Cisco peut avoir besoin afin de diagnostiquer un problème ou classer un rapport de bug. Ceci inclut les vidages mémoire de mémoire, la sortie d'informations de routage, et la **sortie de commande show de** périphérique.

Standardisation de configuration

Les standards de configuration globaux de périphérique représentent la pratique de mettre à jour les périphériques similaires standard et les services de paramètres de configuration globale à travers ayant pour résultat la cohérence au niveau de l'entreprise de configuration globale. Les commandes de configuration globale sont des commandes qui s'appliquent au périphérique entier et pas aux ports individuels, aux protocoles, ou aux interfaces. Les commandes de configuration globale affectent généralement l'accès au périphérique, le comportement général de périphérique, et la sécurité des périphériques. Dans le Cisco IOS ceci inclut des commandes de service, des commandes IP, des commandes vty, des commandes de port de console, se connectant des commandes, des commandes AAA/TACACS+, des commandes SNMP, et des commandes de bannière. Également important dans des standards de configuration globaux de périphérique est un périphérique approprié nommant la convention qui permet à des administrateurs pour identifier le périphérique, le type de périphérique, et l'emplacement de périphérique basé sur le nom de Système de noms de domaine (DNS) du périphérique. La cohérence de configuration globale est importante pour la prise en charge et la fiabilité globales d'un environnement de réseau parce qu'elle aide à réduire la complexité de réseau et à améliorer la prise en charge de réseau. La difficulté de support est souvent expérimentée sans standardisation de configuration due au comportement incorrect ou contradictoire de périphérique, à l'accès SNMP, et à la sécurité des périphériques générale.

La mise à jour des standards de configuration globaux de périphérique est normalement accomplie par un groupe interne d'ingénierie ou d'exécutions qui crée et met à jour des paramètres de configuration globale pour les périphériques semblables de réseau. Il est dans également une bonne pratique de fournir une copie du fichier de configuration globale dans des répertoires TFTP de sorte qu'ils puissent être au commencement téléchargés à tous les

périphériques nouvellement provisionnés. Également utile est un fichier accessible de Web qui fournit au fichier de configuration standard une explication de chaque paramètre de configuration. Quelques organismes configurent même globalement les périphériques semblables sur une base périodique pour aider à assurer la cohérence de configuration globale ou à passer en revue périodiquement des périphériques pour les normes correctes de configuration globale. Protocol et les normes de configuration d'interface représentent la pratique des normes de mise à jour pour la configuration d'interface et de protocole.

Protocol et la cohérence de configuration d'interface améliore la Disponibilité de réseau en réduisant la complexité de réseau, en fournissant le comportement prévu de périphérique et de protocole, et en améliorant la prise en charge de réseau. Protocol ou l'incohérence de configuration d'interface peut avoir comme conséquence le comportement inattendu de périphérique, les questions de routage de trafic, les problèmes accrus de Connectivité, et le temps accru de support réactif. Les normes de configuration d'interface devraient inclure des descripteurs d'interface de CDP, cachant la configuration, et d'autres normes de particularité de protocole. Les standards de configuration spécifiques de Protocol peuvent inclure :

- Configuration de Routage IP
- Configuration DLSW
- Configuration de liste d'accès
- Configuration atmosphère
- Configuration de Relais de trames
- Configuration de spanning tree
- Affectation et configuration VLAN
- Agrégation virtuelle Protocol (VTP)
- HSRP

Remarque: Il est possible d'avoir d'autres standards de configuration spécifiques de protocole selon ce qui est configuré dans le réseau.

Un exemple des normes IP peut inclure :

- Taille de sous-réseau
- L'espace d'adresse IP utilisé
- Protocole de routage utilisé
- Configuration de protocole de routage

La mise à jour des normes de protocole et de configuration d'interface est normalement la responsabilité des groupes d'ingénierie des réseaux et d'implémentation. Le groupe de construction devrait être responsable d'identifier, de tester, de valider, et de documenter les normes. Le groupe d'implémentation est alors responsable d'utiliser les documents ou les modèles de configuration d'ingénierie pour provision de nouveaux services. Le groupe de construction devrait créer la documentation sur tous les aspects des normes prévues pour assurer la cohérence. Des modèles de configuration devraient également être créés pour aider à imposer les standards de configuration. Des groupes d'exécutions devraient également être formés sur les normes et devraient pouvoir identifier les questions non standard de configuration. La cohérence de configuration est de grande assistance pendant le test, la validation, et la phase de certification. En fait, sans modèles de configuration normalisés, il est presque impossible convenablement de tester, valider, ou certifier une version de Cisco IOS pour un réseau modérément grand.

[Disponibilité de Gestion](#)

La Disponibilité de Gestion est le processus de l'amélioration de la qualité utilisant la Disponibilité de réseau comme mesure d'amélioration de la qualité. Beaucoup d'organismes mesurent maintenant le type de Disponibilité et de panne. Les types de panne peuvent inclure le matériel, le logiciel, le lien/transporteur, l'alimentation/environnement, la conception, ou l'erreur utilisateur/processus. En identifiant des pannes et en exécutant l'analyse de la cause d'origine juste après la reprise, l'organisation peut identifier des méthodes pour améliorer la Disponibilité. Presque tous les réseaux qui ont réalisé la Haute disponibilité ont un certain procédé d'amélioration de la qualité.

[Annexe A - Aperçu des versions de Cisco IOS](#)

La stratégie de version logicielle de Cisco IOS est établie autour du développement de logiciel sain, de l'assurance qualité, et de l'arrivée rapide sur le marché, qui est fondamental au succès des réseaux des clients de Cisco.

Le processus est défini autour de quatre catégories de releases, qui sont expliquées ci-dessous :

- Version de déploiement anticipé (ED) (ED)
- Version principale
- Version de déploiement limité (LD)
- Version de déploiement général (GD)

Cisco crée et met à jour un [calendrier de versions d'IOS](#) qui a des informations sur différentes releases, des cibles, des chemins de migration, de nouvelles descriptions de caractéristiques, et ainsi de suite.

La figure ci-dessous montre le cycle de vie de version logicielle de Cisco IOS :

Version de déploiement anticipé (ED)

Les versions d'ED de Cisco IOS sont des véhicules qui apportent la nouveauté au marché. Chaque révision de maintenance d'un version de déploiement anticipé (ED) inclut non seulement des correctifs de bogue, mais également un ensemble de nouvelles caractéristiques, le nouveau support de plate-forme, et les améliorations générales aux protocoles et à l'infrastructure de Cisco IOS. Chacun à deux ans, les caractéristiques et des Plateformes des version de déploiement anticipé (ED) sont mis en communication au prochain se piquent la release de Cisco IOS.

Il y a quatre types de version de déploiement anticipé (ED), chacun avec un modèle de version légèrement différent et étapes importantes du cycle de vie. Les version de déploiement anticipé (ED) peuvent être classifiés en tant que :

- **Releases consolidées d'Early Deployment de technologie (CTED)** — le nouveau modèle de distribution de Cisco IOS utilise le suite de versions déploiement anticipé (ED) consolidé, également connu sous le nom de série « T », d'introduire de nouvelles caractéristiques, nouvelles plates-formes matérielles, et d'autres améliorations au Cisco IOS. Ils s'appellent la technologie consolidée parce qu'ils dépassent les unités commerciales internes (BU) et la branche d'activités des définitions (LOB). Les exemples des versions de la technologie consolidées sont le Cisco IOS 11.3T, 12.0T, et 12.1T.
- **Releases spécifiques d'Early Deployment de technologie (STED)** — les versions STED ont des caractéristiques d'engagement de caractéristique similaire pendant que CTED relâche sauf qu'ils visent une région spécifique de technologie ou de marché. Ils sont toujours

relâchés sur les Plateformes spécifiques et sont seulement sous la direction d'un BU de Cisco. Des versions STED sont identifiées utilisant deux lettres ajoutées à la version majeure. Les exemples des versions STED sont le Cisco IOS 11.3NA, 11.3MA, 11.3WA, et 12.0DA.

- **Le marché spécifique Early Deployment (SMED) libère** — le Cisco IOS SMEDs est différencié de STEDs par le fait qu'ils visent un segment spécifique de marché vertical (ISP, des entreprises, des institutions financières, des sociétés de Telcom, et ainsi de suite). SMEDs incluent des exigences de fonctionnalité spécifiques de technologie seulement pour les Plateformes spécifiques de l'importance utilisées par le marché vertical destiné. Ils peuvent être différenciés de CTEDs par le fait qu'ils sont seulement construits pour les Plateformes spécifiques d'importance pour le marché vertical, tandis que CTEDs serait construit pour plus de Plateformes basées sur une plus large condition requise de technologie. Des releases du Cisco IOS SMED sont identifiées par une lettre ajoutée à la version majeure (juste comme le CTED). Les exemples de SMEDs sont le Cisco IOS 12.0S et 12.1E.
- **Les version de déploiement anticipé (ED) de courte durée, également connus sous le nom de releases X (XED)** — des releases de version XED de Cisco IOS introduit le nouveau matériel et les Technologies au marché. Ils ne fournissent pas des révisions de maintenance logicielle ni ils fournissent des révisions intermédiaires de logiciel normal. Si un défaut est trouvé dans le XED avant sa convergence avec le CTED, une reconstruction de logiciel est initiée et un nombre est ajouté au nom. Par exemple, les Cisco IOS versions 12.0(2)XB1 et 12.0(2)XB2 sont des exemples des reconstructions 12.0(2)XB.

Versions principales

Les versions principales sont les véhicules primaires de déploiement pour des logiciels de Cisco IOS. Ils sont gérés par la division technologique IOS isco de C et consolident des caractéristiques, des Plateformes, la fonctionnalité, la technologie, et la prolifération d'hôte des version de déploiement anticipé (ED) précédents. Stabilité et qualité de recherche de versions principales de Cisco IOS une plus grandes. Pour cette raison, les versions principales ne reçoivent pas l'ajout des caractéristiques ou des Plateformes. Chaque révision de maintenance fournit des correctifs de bogue seulement. Par exemple, les versions du logiciel Cisco IOS 12.1 et 12.2 sont des versions principales.

Les versions principales ont des mises à jour de maintenance planifiée appelées les releases de maintenance qui sont entièrement régression testée, incorporent les correctifs de bogue les plus récents, et ne prennent en charge aucune nouvelle Plateformes ou caractéristique. Le numéro de version d'une version principale identifie la version principale et son niveau de maintenance. Dans le Logiciel Cisco IOS version 12.0(7), 12.0 est le nombre de la version principale, et 7 est son niveau de maintenance. Le numéro de version complet est 12.0(7). De même, 12.1 est une version principale et 12.1(3) est la troisième version de maintenance du principal Logiciel Cisco IOS version 12.1.

Releases du déploiement limité (LD)

Le LD est la phase de la maturité de Cisco IOS entre la FCS et le déploiement général pour des versions principales. Les versions d'ED de Cisco IOS vivent seulement pendant la phase de déploiement limité parce qu'elles n'atteignent jamais la certification GD.

Releases générales du déploiement (GD)

À un moment donné au cours du cycle de vie de la version, Cisco déclarera une version principale pour être prête pour la certification GD. Seulement une version principale peut réaliser l'état GD. Il rencontre l'échéance de certification GD quand Cisco est satisfait que la release a été :

- Avéré par l'exposition étendue du marché dans les réseaux divers.
- Qualifié par des mesures analysées pour des tendances de stabilité et de bogue.
- Qualifié par des enquêtes de satisfaction du client.
- Une réduction de la tendance normale du client fondent des défauts dans la release au-dessus des quatre releases de maintenance précédentes.

Une équipe croix-fonctionnelle de certification de la représentation des clients GD composée d'ingénieurs TAC, d'ingénieurs avancés des services techniques (AES), d'ingénierie de test de système, et de construction de Cisco IOS est formée pour évaluer chaque défaut exceptionnel de la release. Cette équipe donne l'approbation finale pour la certification GD. Une fois une release atteint l'état GD, chaque révision ultérieure de la release est également GD. En conséquence, une fois une release est GD avoué ; il entre dans automatiquement la phase de maintenance restreinte. Tandis que dans cette phase, la construction de la modification du code, y compris des correctifs de bogue avec la principale reprise de code, est strictement limitée et contrôlée par un gestionnaire de programme. Ceci s'assure qu'aucune bogue défavorable n'est introduite à une version de logiciel GD-certifiée de Cisco IOS. GD est réalisé par une version de maintenance particulière. Les mises à jour de maintenance ultérieure pour cette release sont également des releases GD. Par exemple, le Logiciel Cisco IOS version 12.0 a obtenu la certification GD à 12.0(8). Ainsi, les versions du logiciel Cisco IOS 12.0(9), 12.0(10), sont etc des versions GD.

Images expérimentales ou diagnostiques

Des images expérimentales ou diagnostiques désigné parfois sous le nom des offres spéciales d'ingénierie et sont seulement créées quand des problèmes logiciels essentiels ont été identifiés. Ces images ne sont pas une partie du processus de version normal. Les images dans cette catégorie sont des constructions spécifiques de client conçues pour aider à diagnostiquer un problème, pour tester un correctif de bogue, ou pour fournir une difficulté immédiate. Une difficulté immédiate peut être fournie quand ce n'est pas une option d'attendre la prochaine release d'intérim ou de maintenance. Des images expérimentales ou diagnostiques peuvent être établies sur n'importe quelle base prise en charge de logiciel comprenant la maintenance ou les versions intermédiaires de n'importe quel type de release. Aucun fonctionnaire nommant des conventions n'existent, mais dans de nombreux cas le développeur ajoutera des initiales, l'exp (pour expérimental), ou des chiffres supplémentaires au nom d'image de base. Ces images sont seulement prises en charge sur une base provisoire, en même temps que le développement de Cisco, parce que les exécutions de la release de Cisco TAC et de Cisco IOS ne met pas à jour la documentation d'aide telle que des tables des symboles, ou l'historique de base d'image. Ces images ne subissent aucun test interne de Cisco.

[Échéances de cycles de vie de la version](#)

À un certain point, des releases GD sont remplacées par de plus nouvelles releases par les dernières technologies de mise en réseau. Par conséquent, un processus de retraite de release a été établi avec les trois principales échéances suivantes :

- **Fin des ventes (EOS)** — Pour des versions principales, la date EOS est de trois ans après la première date commerciale de l'expédition (FCS). Ceci fixe une date limite l'où la release peut être achetée pour de nouveaux systèmes. La release EOS continue à être disponible pour télécharger du Cisco Connection Online (CCO) pour des mises à niveau de maintenance.
- **Fin de l'ingénierie (EOE)** — La release EOE est la dernière version de maintenance pour la release GD, et suit typiquement environ trois mois après que la release EOS. Les clients peuvent continuer à recevoir le Soutien technique de Cisco TAC, aussi bien que téléchargent la release EOE de CCO. Le bulletin de produit annonçant les releases EOS et EOE et des

dates sont édités un an avant la date prévue EOS. À ce moment, les clients devraient commencer à étudier améliorer leur logiciel de Cisco IOS pour tirer profit des dernières technologies de mise en réseau.

- **Fin de la vie (EOL)** — À la fin du cycle de vie de la version, tout le soutien de la version logicielle de Cisco IOS est terminé et plus disponible pour télécharger la date EOL. Généralement la date EOL est de cinq ans après la date EOE. Un bulletin de produit EOL est édité approximativement un an avant la date de l'effectif EOL.

Version de Cisco IOS nommant la convention

Le convention de dénomination des images de Cisco IOS fournit un profil complet de toutes les images libérées. Le nom inclut toujours l'identifiant de version principale et l'identifiant de release de maintenance. Le nom peut également les reconstruction-identifiants spécifiques inclure un indicateur de série, un indicateur de reconstruction (pour la release de maintenance), des indicateurs de caractéristique spécifique d'unité commerciale (BU), et de BU caractéristique-indicateur. Le format peut être décomposé comme suit :

Nom mer la secti on de conv entio n	Explication
x.y	Une combinaison de deux (un ou deux) identifiants distincts de chiffre a séparé par « . » cela identifie la valeur de version principale. Cette valeur est déterminée par la commercialisation de Cisco IOS. Exemple : 12.1
z	Un à trois chiffres qui identifie la release de maintenance de x.y. Ceci se produit toutes les huit semaines. Les valeurs sont 0 à bêta, 1 à la FCS, et 2 pour la première release de maintenance. Exemple : 12.1(2)
p	Un alpha caractère qui identifie une reconstruction de x.y (z). Les débuts de valeur avec une lettre minuscule « a » pour la première reconstruction, puis « b », et ainsi de suite. Exemple : 12.1(2a)
A	Une à trois alpha lettres sont l'indicateur de la série de versions et sont obligatoires pour des releases CTED, STED, et X. Il identifie également une famille des Produits ou des Plateformes. Lettres de l'utilisation deux de version de déploiement anticipé (ED) de technologie. La première lettre représente la technologie et la deuxième lettre est utilisée pour la différenciation. Exemple : A = Access Server/Dial technology (example:11.3AA) B = Broadband (example:12.2B)

	<p>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 » en la première position du nom de version identifie une release une fois basée sur série CTED la « T ». Par exemple, XA, XB, XC, et ainsi de suite. Un « x » ou « Y » en la deuxième position du nom de version identifie une version ED de courte durée basée en fonction, ou affiliée à, une version STED. Par exemple, 11.3NX (basé sur 11.3NA), 11.3WX (basé sur 11.3WA), et ainsi de suite.</p>
o	<p>Un ou deux indicateur numérique facultatif de chiffre qui identifie une reconstruction d'une valeur particulière de release. Blanc de congé sinon représentant une reconstruction. Débuts avec 1, puis 2, et ainsi de suite. Exemple : 12.1(2)T1, 12.1(2)XE2</p>
u	<p>Un ou deux indicateur numérique de chiffre qui identifie la fonctionnalité de la release de Bu-particularité. La valeur est déterminée par l'équipe de vente de BU. Exemple : 11.3(6)WA4, 12.0(1)W5</p>
v	<p>Un à l'indicateur numérique à deux chiffres qui identifie la release de maintenance du code de Bu-particularité. Les valeurs sont 0 à bêta, 1 à la FCS, et 2 comme première release de maintenance. Exemple : 11.3(6)WA4(9), 12.0(1)W5(6)</p>
p	<p>Un alpha indicateur de caractère qui identifie une reconstruction d'une version de la technologie spécifique. Les débuts de valeur avec une lettre minuscule « a » pour la première reconstruction, puis « b », et ainsi de suite. Exemple : 11.3(6)WA4(9a) serait une reconstruction de 11.3(6)WA4(9).</p>

Le graphique suivant étiquette les différentes sections du Cisco IOS nommant la convention :

[Annexe B - Fiabilité de Cisco IOS](#)

La fiabilité de Cisco IOS est une zone où Cisco tâche continuellement de s'améliorer. Avant de discuter des pratiques recommandées adaptées aux besoins du client, de la compréhension des efforts internes de qualité et de fiabilité IOS de Cisco est nécessaire. Ces sections sont principalement destinées pour fournir un aperçu des efforts plus récents de Cisco de qualité logicielle de Cisco IOS et quelles suppositions de client devraient être faites concernant la fiabilité de logiciel.

Programme de qualité de Cisco IOS

Cisco a un processus de développement bien défini IOS appelé la grande méthodologie d'ingénierie de GEM (GEMME). Ce processus a un cycle de vie triphasé :

- Stratégie et planification
- Exécution
- Déploiement

Les zones générales dans le cycle de vie incluent la hiérarchisation d'introduction de caractéristique, le développement, le processus de test, les phases d'introduction de logiciel, le premier client expédié (FCS), le GD, et l'ingénierie soutenante. Cisco suit également un certain nombre d'instructions de meilleure pratique de qualité logicielle des organismes tels que l'organisation internationale de normalisation (OIN), le Telcordia (autrefois Bellcore), l'IEEE et l'institut de génie logiciel de Carnegie Mellon. Ces instructions sont incorporées aux processus de la GEMME de Cisco. Les processus de développement de logiciel de Cisco sont OIN 9001 (1994) certifiée.

Le procédé primaire pour l'amélioration de qualité logicielle de Cisco IOS est un processus piloté par client par lequel Cisco écoute des clients, définit des buts et des mesures, implémente des pratiques recommandées, et surveille des résultats. Une équipe croix-organisationnelle qui est commise à améliorer la qualité logicielle conduit ce processus. Un diagramme du procédé d'amélioration de la qualité de Cisco IOS est affiché ci-dessous :

Le procédé d'amélioration de la qualité a des buts mesurables distincts pour FY2002 et au-delà. Le centre primaire de ces buts est de réduire des défauts en identifiant des problèmes logiciels plus tôt dans le cycle de test, pour réduire l'arriéré de défaut, pour améliorer la cohérence de caractéristique et la clarté de version logicielle, et pour fournir à calendriers de lancement et à qualité logicielle prévisibles cohérents. Les initiatives pour adresser ces zones des améliorations de test de régression incluent de nouveaux outils de couverture de test (identifiant des zones d'une couverture plus faible de test), amélioration du processus d'action corrective de test, et de Cisco IOS système. Des ressources supplémentaires ont été appliquées pour aborder ces questions et il y a l'engagement exécutif et croix-fonctionnel pour toutes les versions logicielles primaires de Cisco IOS.

Test de version de Cisco IOS

Une partie intégrante de l'effort de qualité de fiabilité de logiciel au sein de Cisco est la qualité, la portée, et la couverture du test. De façon générale, Cisco a les objectifs de qualité suivants IOS :

- Réduisez les défauts internes trouvés de régression de Cisco. Ceci inclut plus de haute qualité à l'étude et l'identification de plus de problèmes dans analyse statique/dynamique.
- Réduisez les défauts trouvés par client
- Réduisez les défauts exceptionnels totaux
- Augmentez la clarté de version logicielle et la cohérence de caractéristique
- Fournissez aux releases de caractéristique et de maintenance des programmes et la qualité

Le test interne de Cisco peut être considéré comme processus où différents défauts sont identifiés dans différentes étapes du test. Le but global est de trouver les bons genres de défauts dans le laboratoire droit. C'est important pour plusieurs raisons. Le premier et le plus important est que la couverture adéquate de test peut ne pas exister à des étapes plus tardives de test. Les coûts de test augmentent également considérablement de l'étape pour présenter en raison de la capacité d'automatiser aux parties et la complexité et l'expertise croissantes exigées plus tard. Le

diagramme suivant dépeint le spectre de test pour le Cisco IOS.

La première phase est développement de logiciel. Cisco a plusieurs efforts dans cette zone d'aider à améliorer la qualité logicielle initiale. Les groupes de développement exécutent également des examens de code ou même de plusieurs examens de code pour s'assurer que d'autres développeurs approuvent des modifications de logiciel ou le nouveau code de caractéristique.

La prochaine étape est test d'unité. Le test d'unité utilise les outils qui examinent l'interaction de logiciel sans utilisation d'un laboratoire. DevTest sont des essais en laboratoire qui incluent le test de caractéristique/fonctionnalité et le test de régression. Le test de caractéristique/fonctionnalité est conçu pour examiner la fonctionnalité d'une caractéristique donnée. Ceci inclut la configuration, la De-configuration, et le test de toutes les permutations de caractéristique comme défini dans la spécification de caractéristique. Le test de régression est fait dans un moyen de tests automatisé conçu pour valider la fonctionnalité et le comportement de caractéristique sur une base actuelle. Le test est concentré principalement sur le routage, la commutation, et la fonctionnalité de caractéristique en un certain nombre de différentes topologies du réseau utilisant les pings et la génération limitée du trafic. Le test de régression est seulement fait sur une combinaison limitée des caractéristiques, des Plateformes, des versions de logiciel, et des topologies dues au nombre extrême de permutations possibles, toutefois plus de 4000 tests de régression des scripts sont utilisés aujourd'hui. Le test d'intégration est conçu pour examiner des capacités d'essai en laboratoire pour plus d'ensemble complet de Produits et d'Interopérabilité. Le test d'intégration augmente également la couverture de code de test en développant le test pour inclure des tests d'Interopérabilité, effort et des tests de performance, des tests de système, et test négatif (événements inattendus de test).

La phase suivante de laboratoire fournit le test de bout en bout pour les environnements communs de client. Ceux-ci sont affichés dans le diagramme ci-dessus en tant que le laboratoire de test financier (FTL) et NSITE, test de scénario de client. Le FTL a été établi pour fournir le test pour la communauté financière essentielle de mission. NSITE est un groupe qui fournit un test plus en profondeur pour différentes Technologies Cisco IOS. Les laboratoires NSITE et FTL se concentrent sur des zones telles que l'évolutivité et les essais de performances, l'évolutivité, la Disponibilité et la résilience, l'Interopérabilité, et l'utilité. L'utilité se concentre sur les questions, la gestion d'événement/corrélation et le dépannage en vrac de ravitaillement sous le chargement. D'autres laboratoires existent au sein de Cisco pour différents marchés verticaux pour aider à tester ces zones.

Le laboratoire final affiché dans le diagramme ci-dessus est identifié comme laboratoire de client. Le test de client est une extension de l'effort de qualité et recommandée pour que les environnements facilement disponibles s'assurent que la combinaison précise des caractéristiques, la configuration, les Plateformes, les modules, et la topologie ont été entièrement testés. La couverture de test devrait inclure l'évolutivité du réseau et la performance dans la topologie identifiée, le test spécifique d'application, le test négatif dans la configuration identifiée, le test d'Interopérabilité pour des périphériques non-Cisco, et le test de brûlure.

[Logiciel MTBF](#)

Une des mesures les plus communes de la fiabilité globale est le temps moyen entre pannes (MTBF). Le MTBF pour la fiabilité de logiciel est utile en raison des capacités d'analyse qui ont été développées pour la fiabilité de matériel utilisant le MTBF. La fiabilité de matériel peut plus exactement être déterminée utilisant quelques normes existantes. Cisco utilise la méthode de compte de pièces basée sur des données standard MTBF des Technologies de Telcordia. Le logiciel MTBF, cependant, n'a aucune méthodologie correspondante d'analyse et doit compter sur

la mesure sur le terrain pour l'analyse MTBF.

Pendant les trois dernières années, Cisco a exécuté des mesures sur le terrain de fiabilité de logiciel pour le réseau informatique interne de Cisco et ce travail est documenté au sein de Cisco. Le travail est basé sur des crash forcés par logiciel pour les périphériques de Cisco IOS, qui peuvent être mesurés utilisant les informations et les informations sur le temps de fonctionnement de déROUTement SNMP de Gestion de réseau. L'étude identifie la fiabilité de logiciel utilisant un modèle log-normal statistique de distribution pour les versions logicielles identifiées. Le temps moyen de réparation (durée moyenne de reprise) de la panne de logiciel est de référence en moyenne reprise et temps de rétablissement de routeur. Six temps de rétablissement minute sont utilisés pour des environnements d'entreprise et quinze minutes est utilisées pour de plus grands fournisseurs d'accès Internet (ISP). Le résultat de cette étude actuelle est que le logiciel atteint généralement la Disponibilité correcte de nines une fois relâché, ou après quelques versions de maintenance, et est encore plus élevé au fil du temps, comme mesuré utilisant le logiciel forcé tombe en panne comme seule source de temps d'arrêt. L'étude a identifié des valeurs potentielles MTBF comme plage entre 5,000 heures pour le logiciel têt de déploiement à 50,000 heures pour le logiciel de déploiement général.

La réfutation la plus commune à ce travail est que les crash forcés par logiciel n'incluent pas tout l'en raison encouru par périodes de panne des questions de fiabilité de logiciel. Si cette mesure est utilisée dans des efforts d'amélioration de la qualité, elle peut aider à améliorer le débit de crash forcés par logiciel mais peut ignorer d'autres zones critiques de fiabilité de logiciel. Ce commentaire demeure dû en grande partie sans réponse à la difficulté dans la fiabilité de logiciel exactement de prévision utilisant une méthodologie statistique. Les statisticiens de qualité logicielle de Cisco ont conclu qu'un plus grand ensemble témoin de données précises serait nécessaire pour prévoir sûrement le logiciel MTBF utilisant une plus grande plage des types de panne. Supplémentaire, l'analyse statistique théorique devrait difficile des variables telles que la complexité de réseau, l'expertise du personnel résoudre des questions connexes de logiciel, la conception de réseaux, les caractéristiques activées, et les procédés de gestion de logiciel.

À ce moment, aucun travail de secteur n'a été terminé à prévoient plus exactement la fiabilité de logiciel avec des mesures sur le terrain dues à la difficulté de collecter exactement ce type de données sensibles. En outre, la plupart des clients mettent ? t veulent Cisco collectant la Disponibilité de l'information directement de leur réseau dû à la nature de propriété industrielle de la Disponibilité de données. Quelques organismes cependant collectent des données sur la fiabilité de logiciel et Cisco encourage des organismes à collecter des mesures sur la Disponibilité due aux pannes de logiciel, et à exécuter l'analyse de la cause d'origine sur ces pannes. Les organismes avec une fiabilité de logiciel plus élevée ont utilisé cette position proactive pour améliorer la fiabilité de logiciel par un certain nombre de pratiques qu'ils peuvent contrôler.

Suppositions de fiabilité de logiciel

En raison du feedback de la clientèle, les études proactives réalisées par l'analyse de groupe et de cause principale de Technologies Cisco IOS exécutée par les Services avancés de Cisco team, quelques plus nouvelles suppositions et des pratiques recommandées ont été formées qui aident à améliorer la fiabilité de logiciel. Ces suppositions portent sur des responsabilités de test, maturité ou âge de logiciel, des caractéristiques activées, et le nombre de versions de logiciel déployées.

Responsabilité de test

La première nouvelle supposition traite la responsabilité de test. Cisco est toujours responsable du

test/validant de nouvelles fonctionnalités et caractéristiques pour s'assurer qu'elles fonctionnent dans des produits nouveaux. Cisco est également responsable du test de régression pour s'assurer que les nouvelles versions de logiciel sont arrière - compatible. Cependant, Cisco ne peut pas valider chaque caractéristique, topologie, et plate-forme contre chaque mise en garde potentielle qu'un environnement de client peut appliquer (les idiosyncrasies de conception, le chargement, et les profils du trafic). Les pratiques recommandées facilement disponibles pour des clients incluent le test en topologie de travaux pratiques réduite qui imite le réseau de production utilisant les caractéristiques, la conception, les services, et le trafic de l'application définis par client.

Fiabilité contre la maturité de logiciel

La fiabilité de logiciel est principalement un facteur de maturité de logiciel. Le logiciel mûrit pendant qu'il reçoit l'exposition (utilisation) et comme des bogues identifiées sont corrigées. Cisco relâchent des exécutions sont allés à une architecture de release de série s'assurer que le logiciel mûrit sans nouvelles caractéristiques étant ajoutées. Les clients qui ont besoin de la Haute disponibilité recherchent un logiciel plus mûr avec les configurations des lesquelles ils ont besoin maintenant. Un compromis existe alors entre la maturité du logiciel, les exigences au niveau de la disponibilité, et les moteurs opérationnels pour de nouvelles caractéristiques ou fonctionnalité. Beaucoup d'organismes ont des normes ou des instructions pour la maturité acceptable. Certains recevront seulement la cinquième version intermédiaire d'une série particulière. Pour d'autres, ce peut être le neuvième ou certification GD. Finalement, les besoins d'organisation de décider leurs taux acceptables de risque en termes de maturité de logiciel.

Fiabilité contre la quantité de caractéristiques et de normes

La fiabilité de logiciel est également un facteur de quelle quantité de code est testé et exercé dans un environnement de production. À mesure que la quantité de différents plates-formes matérielles et modules augmente, la quantité de code exercée également augmente, qui augmente généralement l'exposition aux erreurs de logiciel. Les mêmes peuvent être dits pour la quantité de protocoles configurés, la variété de configurations, et même la variété de topologie ou de conceptions mises en application. La conception, la configuration, les protocoles, et les facteurs de module de matériel peuvent contribuer à la quantité de code qui est exercé et au risque ou à l'exposition accru aux erreurs de logiciel.

Les exécutions de version logicielle ont maintenant un logiciel de but spécifique qui limite généralement le code disponible dans une zone particulière. Les unités commerciales ont recommandé les conceptions et les configurations qui plus complètement sont testées au sein de Cisco et plus largement sont utilisées par des clients. Les clients ont également commencé à adopter des pratiques recommandées pour que des topologies modulaires normalisées et des configurations standard diminuent la quantité d'exposition non essayée de code et d'améliorent la fiabilité de logiciel globale. Quelques réseaux facilement disponibles ont des instructions de configuration de norme stricte, des normes modulaires de topologie, et le contrôle de version de logiciel à aider à réduire le risque d'exposition non essayée de code.

Fiabilité contre le nombre de versions déployées

Un autre facteur de fiabilité de logiciel est Interopérabilité entre les versions et la quantité pure de code qui obtient exercé avec des plusieurs versions. À mesure que la quantité de versions de logiciel augmente, la quantité de code exercée également augmente, qui augmente alors l'exposition aux erreurs de logiciel. Le risque à la fiabilité augmente presque exponentiellement en raison du code supplémentaire exercé avec des plusieurs versions. On l'identifie maintenant que les organismes doivent exécuter au moins une poignée de versions dans le réseau pour couvrir

des conditions requises de caractéristique spécifique et de plate-forme. S'exécuter plus de cinquante versions dans un environnement de réseau en grande partie homogène cependant, est normalement indicatif des problèmes logiciels dus à l'incapacité correctement d'analyser ou valider ce beaucoup de versions.

Pour améliorer la fiabilité de logiciel, le développement de Cisco réalise l'essai de régression de logiciel pour s'assurer que les différentes versions de logiciel sont compatibles. En outre, code logiciel est plus modulaire et les principaux modules sont moins pour changer de manière significative entre les versions au fil du temps. Cisco libèrent des exécutions ont également changé la quantité de logiciel disponible aux clients pendant que des versions avec des défauts connus ou des problèmes d'interopérabilité sont rapidement retirées de CCO pendant que des défauts sont trouvés.

[Informations connexes](#)

- [Systèmes d'exploitation Cisco IOS](#)
- [Support et documentation techniques - Cisco Systems](#)