Cisco IOS-beheer voor netwerken met hoge beschikbaarheid: Whitepaper over beste praktijken

Inhoud

Inleiding

Het implementeren en onderhouden van betrouwbare Cisco IOS®-software is een prioriteit in de bedrijfskritieke netwerkomgeving van vandaag die hernieuwde aandacht van Cisco en de klant vereist om non-stop beschikbaarheid te bereiken. Hoewel Cisco zich moet richten op hun belofte om de kwaliteit van de software te verbeteren, moeten de groepen voor netwerkontwerp en ondersteuning zich ook richten op de beste praktijken voor Cisco IOS-softwarebeheer. Het doel is een grotere beschikbaarheid en efficiëntie van softwarebeheer. Deze methode is een gecombineerd partnerschap om beste praktijken voor softwarebeheer te delen, te leren en te implementeren.

Dit document biedt een effectief operationeel raamwerk van Cisco IOS-beheerpraktijken voor zowel Enterprise- als Serviceprovider-klanten die helpen bij het promoten van een verbeterde softwarebetrouwbaarheid, een verminderde netwerkcomplexiteit en een verhoogde netwerkbeschikbaarheid. Dit kader helpt ook de efficiëntie van softwarebeheer te verbeteren door gebieden van verantwoordelijkheid en overlappingen te identificeren in testen en valideren van softwarebeheer tussen Cisco release bewerkingen en de Cisco Customer Base.

Overzicht van Cisco IOS-beste praktijken

De volgende tabellen bieden een overzicht van de Cisco IOS-beste praktijken. Deze tabellen kunnen worden gebruikt als een beheeroverzicht van de gedefinieerde optimale werkwijzen, een controlelijst voor lacuneanalyse om de huidige Cisco IOS-beheerpraktijken te bekijken, of als een kader voor het maken van processen rond Cisco IOS-beheer.

De tabellen definiëren de vier levenscycluscomponenten van Cisco IOS-beheer. Elke tabel begint met een strategie en een samenvatting van de instrumenten voor het geïdentificeerde levenscyclusgebied. Na de strategie en de gereedschapswerktuigen zijn er specifieke beste werkwijzen die alleen van toepassing zijn op het afgebakende levenscyclusgebied.

Planning - Bouwen aan het kader van het Cisco IOS Beheer - Planning is de eerste fase van Cisco IOS het beheer dat nodig is om een organisatie te helpen bepalen wanneer u software wilt upgraden, waar te en wat proces zal worden gebruikt om potentiële beelden te testen en valideren.

Best Practice	Detail
Strategie en tools voor Cisco IOS-planning	Om te beginnen met Cisco IOS beheerplanning begint met een eerlijke beoordeling van huidige praktijken, de ontwikkeling van haalbare doelstellingen en projectplanning.
Softwareversie en -definities	Geeft aan waar softwareconsistentie kan worden gehandhaafd. Een software track kan worden gedefinieerd als een unieke softwareversiegroep, gedifferentieerd van andere gebieden naar unieke geografie, platforms, module of functievereisten.
Upgradeprogramma en -definities	De definities van de upgrade-cyclus kunnen worden gedefinieerd als basiskwaliteitsstappen in de software en het wijzigingsbeheer, die worden gebruikt om te bepalen wanneer een software-

	upgradecyclus moet worden gestart.
certificeringsproces	De certificeringsprocesstappen dienen onder meer betrekking te hebben op de identificatie van het spoor, de definities van de upgradecycli, het beheer van de kandidaat, de beproeving/validatie en ten minste enig gebruik van de proefproductie.

Design - Selectie en validatie van IOS versies - Het hebben van een goed gedefinieerd proces voor het selecteren en valideren van Cisco IOS versies helpt een organisatie om ongeplande downtime door onsuccesvolle upgradepogingen en ongeplande softwaredefecten te verminderen.

Best Practice	Detail
Strategie en tools voor Cisco IOS selectie en validatie	Defineert processen voor het selecteren, testen en valideren van nieuwe Cisco IOS-versies. Dit omvat een netwerk testlaboratorium dat het productienetwerk emuleert
Kandidaatbeheer	Candidate management is het identificeren van vereisten voor softwareversie en mogelijke risico's voor de specifieke hardware en enabled-functiesets.
Testen en valideren	Testen en valideren is een cruciaal aspect van softwarebeheer en een netwerk met hoge beschikbaarheid. Correct laboratoriumtesten kunnen de productie-onderbreking aanzienlijk beperken, het ondersteunend personeel van het netwerk helpen trainen en helpen bij het stroomlijnen van de processen voor netwerkimplementatie.

Implementatie - Snelle en Succesvolle Cisco IOS implementatieprocessen - Goed gedefinieerd implementatieprocessen staan een organisatie toe om snel en met succes nieuwe Cisco IOS-versies te implementeren.

Best Practice	Detail
Strategie en tools voor Cisco IOS-implementaties	De basisstrategie voor Cisco IOS-implementaties is om definitieve certificering via een proefproces en snelle implementatie uit te voeren met behulp van upgradetools en een goed gedefinieerd implementatieproces.
proefproces	Om de potentiële blootstelling tot een minimum te beperken en eventuele resterende productieproblemen op een veiliger wijze op te vangen, wordt een

	softwarepiloot aanbevolen. In het individuele proefplan moeten proefselectie, proefduur en meting worden overwogen.
Uitvoering	Na de voltooiing van de proeffase moet de Cisco IOS-implementatiefase beginnen. De implementatiefase kan meerdere stappen omvatten om te zorgen voor succes en efficiëntie van software, waaronder langzame start, definitieve certificering, upgradevoorbereiding, upgradeautomatisering en definitieve validatie.

Transacties - Beheerd de Hoge beschikbaarheid van Cisco IOS Implementatie-Beste praktijken voor Cisco IOS operaties omvatten de controle van de softwareversie, Cisco IOS het Syrische beheer, probleembeheer, configuratie standaardisatie, en beschikbaarheidsbeheer.

Best Practice	Detail
Strategieën en tools voor Cisco IOS-bewerkingen	De eerste strategie van Cisco IOS operaties is het milieu zo eenvoudig mogelijk te houden, vermijdend variatie in configuratie en Cisco IOS versies. De tweede strategie is de mogelijkheid om netwerkfouten te identificeren en snel op te lossen.
Software versie Control	Softwareversiecontrole is het proces van het implementeren van alleen gestandaardiseerde softwareversies en het controleren van het netwerk om software te valideren of mogelijk te wijzigen als gevolg van de niet-versie-conformiteit.
Proactief systeembeheer	De inzameling, controle, en analyse van Syslog zijn de processen van het foutbeheer die worden aanbevolen om meer IOS specifieke netwerkproblemen op te lossen die moeilijk of onmogelijk om met andere middelen te identificeren zijn.
Probleembeheer	Gedetailleerde probleembeheerprocessen die probleemidentificatie, informatieverzameling en een goed geanalyseerd oplossingspad definiëren. Deze gegevens kunnen worden gebruikt om de oorzaak te bepalen.
Standaardisatie voor configuratie	Configuratienormen vormen de praktijk van het creëren en onderhouden van standaard mondiale configuratieparameters voor soortgelijke apparaten en diensten, die resulteren in wereldwijde configuratie consistentie voor het gehele bedrijf.

Beschikbaarheidsbeheer	Beschikbaarheidsbeheer is het proces van kwaliteitsverbetering door gebruik te maken van netwerkbeschikbaarheid als de kwaliteitsverbetering.

Overzicht van het beheerproces van software

Cisco IOS-beheer van de levenscyclus van software is gedefinieerd als de reeks planning, ontwerp, implementatie en operationele processen die worden aanbevolen voor betrouwbare softwareimplementaties en netwerken met een hoge beschikbaarheid. Dit omvat processen om Cisco IOS-versies in het netwerk te selecteren, te valideren en te onderhouden.

Het doel van het beheer van de levenscyclus van Cisco IOS-software is de netwerkbeschikbaarheid te verbeteren door de waarschijnlijkheid van productie-geïdentificeerde softwaredefecten of software-gerelateerde verandering/upgrademislukkingen te verminderen. De beste praktijken die binnen deze documentatie worden gedefinieerd zijn vertoond om dergelijke tekortkomingen en veranderingsfouten te verminderen op basis van de praktische ervaring van veel klanten van Cisco en het Cisco Advanced Services team. Het beheer van de levenscyclus van software kan aanvankelijk de uitgaven verhogen, maar lagere totale kosten van eigendom kunnen worden gerealiseerd door minder tekorten en meer gestroomlijnde implementatie en ondersteuningsmechanismen.

Planning - Bouwen aan het Cisco IOS-beheerframework

Planning is de eerste fase van Cisco IOS beheer dat nodig is om een organisatie te helpen bepalen wanneer u software wilt upgraden, waar en wat proces zal worden gebruikt om potentiële beelden te testen en valideren.

De beste praktijken zijn onder meer softwareversie, upgradecyclus en definities, en de invoering van een interne softwarecertificatie.

Strategie en tools voor Cisco IOS-planning

Begin met Cisco IOS beheerplanning met een eerlijke beoordeling van huidige praktijken, de ontwikkeling van haalbare doelstellingen en projectplanning. Een zelfbeoordeling moet plaatsvinden door de beste praktijken in dit document te vergelijken met processen binnen uw organisatie. De fundamentele vragen moeten het volgende omvatten:

- Heeft mijn organisatie een softwarecertificeringsproces dat ook softwaretesten/validatie omvat?
- Heeft mijn organisatie Cisco IOS softwarereleases met een beperkt aantal Cisco IOS versies die in het netwerk lopen?
- Heeft mijn organisatie moeite om te bepalen wanneer u Cisco IOS-software wilt upgraden?
- Heeft mijn organisatie problemen met het implementeren van nieuwe Cisco IOS-software, zowel efficiënt als effectief?
- Heeft mijn organisatie Cisco IOS stabiliteitskwesties na plaatsing die de kosten van onderbreking ernstig beïnvloeden?

Na de evaluatie moet uw organisatie beginnen met het definiëren van doelstellingen voor Cisco IOS-softwarebeheer. Start door een cross-functionele groep managers en/of lopen van

architectuurplanningsgroepen, engineering, implementatie en bewerkingen bijeen te brengen om Cisco IOS-doelstellingen en projecten voor procesverbetering te definiëren. Het doel van de eerste bijeenkomsten moet zijn algemene doelstellingen, taken en verantwoordelijkheden vast te stellen, actiepunten toe te wijzen en initiële projectschema's vast te stellen. Bepaal ook kritische succesfactoren en maatstaven om de voordelen van softwarebeheer te bepalen. Mogelijke metriek zijn:

- beschikbaarheid (vanwege softwareproblemen)
- kosten van software-upgrades
- tijd nodig voor upgrades
- aantal softwareversies die in productie zijn
- software-upgrade verandert succes/mislukkingen

Naast de algemene planning van het Cisco IOS beheerkader, definiëren sommige organisaties ook lopende software planning vergaderingen om maand of driemaandelijks te gebeuren. Het doel van deze bijeenkomsten is de huidige softwareimplementatie te evalueren en te beginnen met het plannen van nieuwe softwarevereisten. De planning kan onder meer bestaan uit het herzien of wijzigen van de huidige softwarebeheerprocessen, of simpelweg het definiëren van rollen en verantwoordelijkheden voor de verschillende fasen van het softwarebeheer.

Gereedschappen in de planningsfase bestaan uitsluitend uit instrumenten voor het beheer van de softwareinventaris. De CiscoWorks 2000 Resource Manager Essentials (RME) voorraadbeheer is het primaire gereedschap dat in dit gebied wordt gebruikt. De CiscoWorks2000 manager van de inventaris van Cisco vereenvoudigt zeer het versiebeheer van Cisco routers en switches door op web-gebaseerde rapportagetools die Cisco IOS apparaten op basis van softwareversie, platform, geheugengrootte en apparaatnaam rapporteren en sorteren.

Softwareversie en -definities

De eerste Cisco IOS-softwarebeheerplanning met beste praktijken identificeert waar de softwareconsistentie kan worden gehandhaafd. Een software track is gedefinieerd als een unieke softwareversiegroep, gedifferentieerd van andere gebieden naar unieke geografie, platforms, module of functievereisten. Optimaliseer, zou een netwerk slechts één softwareversie moeten draaien. Dit verlaagt de kosten voor softwarebeheer aanzienlijk en biedt een consistente en gemakkelijk beheerde omgeving. In werkelijkheid moeten de meeste organisaties echter meerdere versies in het netwerk uitvoeren vanwege problemen met functies, platform, migratie en beschikbaarheid binnen specifieke gebieden. In veel gevallen werkt dezelfde versie niet op heterogene platforms. In andere gevallen kan de organisatie niet wachten tot één versie die al hun vereisten ondersteunt. Het doel is de kleinste softwaresporen voor het netwerk te identificeren met inachtneming van test-/validatie-, certificatie- en upgradevereisten. In veel gevallen kan de organisatie iets meer sporen hebben om de kosten voor testen/validatie, certificering en upgrades te verlagen.

Het eerste differentierende feit is platformondersteuning. Meestal hebben LAN-switches, WAN-switches, kernrouters en randrouters afzonderlijke softwaresporen. Er kunnen andere softwaresporen nodig zijn voor specifieke functies of diensten, zoals DLSw-datalink-switching (Data-link Switching), Quality of Service (QoS) of IP-telefonie, vooral als deze eis binnen het netwerk gelokaliseerd kan worden.

Een ander criterium is betrouwbaarheid. Veel organisaties proberen de meest betrouwbare software naar de kern van het netwerk en het datacenter te leiden, terwijl ze nieuwere geavanceerde functies, of hardwareondersteuning, naar de rand bieden. Aan de andere kant, schaalbaarheid of bandbreedte eigenschappen zijn vaak het meest nodig in kern of datacenter

omgevingen. Andere sporen kunnen nodig zijn voor specifieke platforms, zoals grotere distributielocaties die een verschillend WAN routerplatform hebben. De volgende tabel is een voorbeelddefinitie van softwaresporen voor een grote ondernemersorganisatie.

spoor	Gebied	hardware platforms	Functies	Cisco IOS-versie	Certificeringsstatus
1	LAN-kernswitching	6500	QoS	12.1E(A8)	Testen
2	LAN-toegangsswitch	2924XL 2948XL switch	Unidirectional Link Detection Protocol (UDLD), Spanning Tree Protocol (STP)	12.0(5.2)XU	Gecertificeerd 3/1/01
3	LAN-distributie/toegang	5500 6509	supervisor 3	5.4(4)	Gecertificeerd op 7/1/01
4	Distribution switch Switch-module (RSM)	RSM	Open kortste pad eerst (OSPF) routing	12.0(11)	Gecertificeerd 3/4/02
5	WAN-head-end distributie	7505 7507 7204 7206	OSPF-Frame Relay	12.0(11)	Gecertificeerd 11/1/01
6	WAN-toegang	2600	OSPF-Frame Relay	12.1(8)	Gecertificeerd 6/1/01
7	IBM-connectiviteit	3600	Synchronous Data Link Control (SDLC) head-end	11.3(8)T1	Gecertificeerd 11/1/00

Spoorwegopdrachten kunnen ook in de loop der tijd veranderen. In veel gevallen kunnen functies of hardwareondersteuning integreren in meer software versies die verschillende paden mogelijk maken om uiteindelijk samen te migreren. Zodra definities zijn vastgelegd, kan de organisatie andere gedefinieerde processen gebruiken om te migreren naar consistentie en validatie van nieuwe versies. De definities van het spoor zijn ook een voortdurende inspanning. Wanneer een nieuwe functie-, service-, hardware- of modulebehoefte wordt vastgesteld, moet een nieuw spoor worden overwogen.

Organisaties die een spoorproces willen starten, moeten beginnen met nieuw vastgestelde spooreisen, of in sommige gevallen met stabilisatieprojecten voor bestaande netwerken. Een organisatie kan ook een aantal identificeerbare communicatie met bestaande softwareversies hebben die de huidige spoordefinitie mogelijk maken. In de meeste gevallen is een snelle migratie naar geïdentificeerde versies niet vereist als de klant voldoende netwerkstabiliteit heeft. De netwerkarchitectuur, of de technische groep, is doorgaans eigenaar van het "track definition"-proces. In sommige gevallen kan één individu verantwoordelijk zijn voor de definitie van het spoor. In andere gevallen zijn projectleiders verantwoordelijk voor het ontwikkelen van softwarevereisten en nieuwe, op individuele projecten gebaseerde definities. Het is ook een goed idee om de spoordefinities op kwartaalbasis te herzien om vast te stellen of nieuwe sporen nodig zijn, of dat oude sporen moeten worden geconsolideerd of verbeterd.

Organisaties die softwaresporen met strikte versiecontrole identificeren en onderhouden, blijken het grootste succes te hebben met een afnemend aantal softwareversies in het productienet. Dit leidt in het algemeen tot een betere softwarestabiliteit en een algehele betrouwbaarheid van het netwerk.

Upgradeprogramma en -definities

De definities van de upgrade-cyclus zijn gedefinieerd als basiskwaliteitsstappen in de software en het wijzigingsbeheer, die worden gebruikt om te bepalen wanneer een software-upgradecyclus moet worden gestart. Omschrijvingen van upgradeprogramma's maken het voor een organisatie mogelijk om een softwareupgradecyclus goed te plannen en de benodigde middelen toe te wijzen. Zonder upgradecyclusdefinities ervaart een organisatie doorgaans een toename van de betrouwbaarheid van de software door de eisen van de eigenschappen in de huidige stabiele versies. Een andere blootstelling zou kunnen zijn dat de organisatie de kans mist om een nieuwe versie goed te testen en valideren voordat productiegebruik vereist is.

Een belangrijk aspect van deze praktijk is het bepalen wanneer en in welke mate softwareplanningsprocessen moeten worden gestart. Dit is vanwege het feit dat een belangrijke oorzaak van softwareproblemen zich voordoet bij het aanzetten van een functie-, service- of hardwarecapaciteit in productie zonder zorgvuldigheid, of het upgraden naar een nieuwe Cisco IOS-versie zonder overwegingen van softwarebeheer. Een ander probleem is niet de verbetering. Door normale softwarecycli en -vereisten te negeren, staan veel klanten voor de moeilijke taak om software te verbeteren door een aantal verschillende belangrijke releases. De moeilijkheid is te wijten aan beeldformaten, standaard gedragsveranderingen, de veranderingen van het bevelniveau tolk (CLI), en protocol veranderingen.

Cisco raadt een duidelijk gedefinieerde upgradecyclus aan, gebaseerd op de beste praktijken zoals gedefinieerd in dit document, die moet worden gestart wanneer nieuwe belangrijke functies, service of hardwareondersteuning nodig is. De mate van certificatie en beproeving/validatie moet worden geanalyseerd (op basis van risico) om de precieze test-/valideringseisen te bepalen. Een risicoanalyse kan worden uitgevoerd op basis van geografische locatie, logische locatie (kern-, distributie- of toegangslagen) of het geschatte aantal getroffen mensen/klanten. Als de

belangrijkste functie- of hardwarefunctie in de huidige release is opgenomen, moeten ook enkele gestroomlijnde upgrade-cyclusprocessen worden gestart. Als deze optie relatief klein is, neemt u het risico in overweging en besluit u welke processen moeten worden gestart. Daarnaast dient de software binnen twee jaar of minder te worden bijgewerkt om er zeker van te zijn dat uw organisatie relatief actueel blijft en dat het upgradeproces niet te omslachtig is.

Klanten dienen ook rekening te houden met het feit dat er geen bug-oplossingen worden toegepast voor software-treinen die de end-of-life (EOL) status hebben bereikt. Er moet ook enige aandacht worden besteed aan de bedrijfsvereisten, aangezien veel omgevingen meer veelzijdige toevoegingen met weinig of geen test-/valideringsprocessen en een aantal daaruit voortvloeiende downtime kunnen tolereren of zelfs verwelkomen. Klanten moeten ook rekening houden met de nieuwere gegevens die in Cisco-release-bewerkingen zijn verzameld, wanneer ze hun testvereisten overwegen. Een analyse van insecten en worteloorzaken toonde aan dat de overgrote meerderheid van bug root oorzaken het resultaat waren van developers die coderen binnen het getroffen softwaregebied. Dit betekent dat als een organisatie een bepaalde functie of module aan hun netwerk toevoegt in een bestaande release, er de waarschijnlijkheid is van het ervaren van een bug gerelateerd aan die functie of module, maar een veel lagere kans dat de nieuwe functie, hardware of module andere gebieden zal beïnvloeden. Deze gegevens moeten organisaties in staat stellen de testvereisten te verlagen wanneer zij nieuwe functies of modules toevoegen die worden ondersteund door bestaande releases te testen, door alleen de nieuwe dienst of functie te testen in combinatie met andere toegelaten diensten. De gegevens moeten ook in aanmerking worden genomen bij het verbeteren van software op basis van een paar kritische insecten in het netwerk.

In de volgende tabel worden de aanbevolen upgradevereisten voor een grote organisatie met een hoge beschikbaarheid van ondernemingen weergegeven:

Trillingen voor softwarebeheer	Vereisten voor levenscyclus van software
Nieuwe netwerkservice. Bijvoorbeeld een nieuwe ATM backbone of een nieuwe VPN-service.	Volledige validatie van de software van de levenscyclus, met inbegrip van nieuwe functietests (in combinatie met andere gefaciliteerde diensten), ingestort topologie-testen, wat-als-prestatie-analyse, en testen van het toepassingsprofiel.
Nieuwe netwerkmogelijkheden worden niet ondersteund in de huidige softwarerelease. Tot de voorbeelden behoren QoS en Multiprotocol Label Switching (MPLS).	Volledige validatie van de software van de levenscyclus, met inbegrip van nieuwe functietests, in combinatie met andere toegelaten diensten, ineenstortende topologie-testen, wat-als-prestatie-analyse, en testen van het toepassingsprofiel.
Nieuwe belangrijke functie of hardwaremodule die in de huidige release bestaat. Bijvoorbeeld,	Kandidaat management proces. Mogelijke volledige validatie op basis van vrijgavevereisten. Mogelijke beperkte test/validatie indien kandidaat management

het toevoegen van een nieuwe GigE module, multicast steun, of DLSW.	de huidige release als mogelijk aanvaardbaar identificeert.
Minder optie toegevoegd. Bijvoorbeeld een TACACS-apparaat voor toegangscontrole.	Denk eens na over het kandidaat-beheer op basis van het risico van het kenmerk. Overweeg het testen of afwerken van de nieuwe functie op basis van risico.
Software die twee jaar in productie is of een driemaandelijks softwareonderzoek.	Kandidaat management- en bedrijfsbeslissingen met betrekking tot de voltooiing van het levenscyclusbeheer om de huidige supporteerbare release te identificeren.

Noodupgrades

In sommige gevallen worden organisaties geconfronteerd met de noodzaak om de software te verbeteren vanwege catastrofale problemen. Dit kan tot problemen leiden als de organisatie geen verbeteringsmethodologie voor noodgevallen heeft. Problemen met software kunnen variëren van onbeheerde software-upgrades, waar software wordt bijgewerkt zonder beheer van de levensduur van de software, tot situaties waarin netwerkapparaten voortdurend crashen, maar de organisatie geen upgrade uitvoert omdat de certificering/test op de volgende kandidaat-release niet is voltooid. Cisco raadt een noodupgradeproces aan voor deze situaties waar beperkte tests en piloten worden uitgevoerd in minder bedrijfskritieke gebieden van het netwerk.

Als catastrofale fouten zonder duidelijke tijdelijke oplossing voorkomen en het probleem met betrekking tot de software is defect, raadt Cisco aan dat de ondersteuning van Cisco volledig wordt ingeschakeld om het defect te isoleren en vast te stellen of of een tijdelijke oplossing beschikbaar is. Wanneer de oplossing beschikbaar is, raadt Cisco een upgrade-cyclus voor noodgevallen aan om snel te bepalen of het probleem met beperkte downtime kan worden gerepareerd. In de meeste gevallen voert een organisatie een ondersteunde versie van de code uit en de probleemoplossing is beschikbaar in een bestaande nieuwere tussenversie van de software.

De organisaties kunnen zich ook voorbereiden op mogelijke noodupgrades. Voorbereiding omvat de migratie naar ondersteunde Cisco IOS-releases en de identificatie/ontwikkeling van kandidaat-vervangende versies in dezelfde Cisco IOS-trein als de gecertificeerde versie. Ondersteunde software is belangrijk omdat het betekent dat de ontwikkeling van Cisco nog steeds bug-fixes aan de geïdentificeerde softwastrein toevoegt. Door ondersteunde software in het netwerk te onderhouden, verkort de organisatie de validatietijd vanwege de meer bekende en stabiele codebasis. Meestal is een kandidaat-vervanging een nieuw tijdelijk softwarebeeld binnen dezelfde Cisco IOS-trein zonder functies of hardwareondersteuning. Een kandidaat-vervangingsstrategie is met name belangrijk als de organisatie zich in de vroege adoptiefase van een bepaalde softwaretrein bevindt.

certificeringsproces

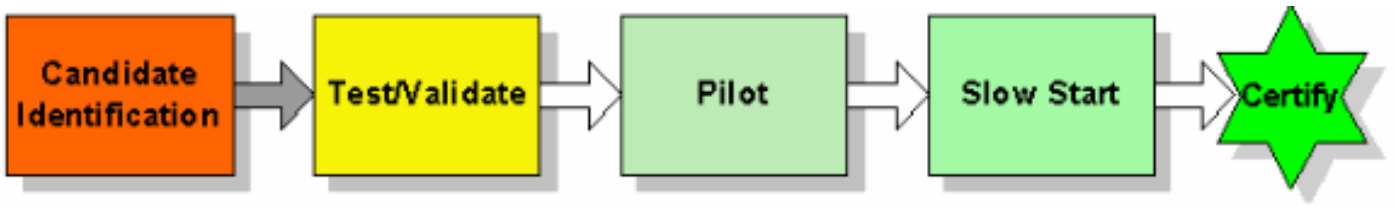
Een certificeringsproces helpt ervoor te zorgen dat gevalideerde software op consistente wijze wordt gebruikt in de productieomgeving van de organisatie. De certificeringsprocesstappen dienen

onder meer betrekking te hebben op de identificatie van het spoor, de definities van de upgradecycli, het beheer van de kandidaat, het testen/valideren en een aantal proefproductietoepassingen. Een eenvoudig certificeringsproces helpt er echter nog steeds voor te zorgen dat consistente softwareversies binnen de geïdentificeerde sporen worden gebruikt.

Start een certificeringsproces door individuen te identificeren die afkomstig zijn van architectuur, techniek/implementatie en operaties om het certificeringsproces op te stellen en te beheren. De groep zou eerst moeten nadenken over bedrijfsdoelstellingen en capaciteitsmogelijkheden om te verzekeren dat het certificeringsproces succesvol zal zijn. Vervolgens wordt de algemene verantwoordelijkheid toegewezen aan personen of groepen voor de belangrijkste stappen in het certificeringsproces, waaronder het beheer van het spoor, de definities van levenscyclusupgrades, de beproeving/validatie en de piloten. Elk van deze gebieden dient binnen de organisatie te worden vastgesteld, goedgekeurd en formeel te worden meegedeeld.

Omvat tevens richtsnoeren voor kwaliteit of goedkeuring in elke fase van het certificeringsproces. Dit wordt soms een "quality gate-proces" genoemd, omdat aan bepaalde kwaliteitscriteria moet worden voldaan voordat het proces naar de volgende stap kan overgaan. Dit helpt ervoor te zorgen dat het certificeringsproces doeltreffend is en de toegewezen middelen waard is. Als kwesties op één gebied met kwaliteit worden gevonden, wordt de inspanning in het algemeen met één stap teruggedrongen.

Softwarekandidaten voldoen misschien niet aan de gedefinieerde certificeringscriteria vanwege de softwarekwaliteit of onverwacht gedrag. Als er problemen worden aangetroffen die invloed hebben op het milieu, moet de organisatie een gestroomlijnder proces hebben om een later tussentijds release te certificeren. Dit helpt de behoefte aan middelen te verminderen en is over het algemeen effectief als de organisatie kan begrijpen wat er is veranderd en welke tekortkomingen zijn opgelost. Het is niet ongebruikelijk voor een organisatie om een probleem met een eerste kandidaat te ervaren en om een later tussentijds Cisco IOS-release te certificeren. Organisaties kunnen ook een beperkte certificering of voorbehouden bieden indien er problemen zijn en kunnen overgaan tot een latere, volledig gecertificeerde release wanneer een nieuwe tijdelijke erkenning is gevalideerd. Het onderstaande stroomschema is een basiscertificeringsproces en omvat kwaliteitsladingen (een beoordeling na elk blok):



Design - Selectie en validatie van Cisco IOS-versies

Een welomschreven methodologie voor het selecteren en valideren van Cisco IOS-versies helpt een organisatie om niet-geplande downtime te reduceren als gevolg van onsuccesvolle upgradepogingen en ongeplande softwaredefecten.

De ontwerpfase omvat het beheer van de kandidaat en het testen/valideren ervan. Candidate management is het proces dat gebruikt wordt om specifieke versies voor de gedefinieerde softwaresporen te identificeren. Testen/valideren maakt deel uit van het certificeringsproces en zorgt ervoor dat de vastgestelde softwareversie binnen het vereiste traject succesvol is. Testen/valideren dient te worden uitgevoerd in een labomgeving met een samengevouwen topologie en configuratie die sterk lijkt op de productieomgeving.

Strategie en tools voor Cisco IOS selectie en validatie

Elke organisatie zou een proces moeten hebben om de standaard IOS versies van Cisco voor het netwerk te selecteren en te valideren die met een proces beginnen voor het selecteren van de Cisco IOS versie. Een functioneel team van architectuur, techniek en operaties moet het proces van het kandidaatbeheer definiëren en documenteren. Na goedkeuring moet het proces worden overgezet naar de geschikte leveringsgroep. Ook wordt aanbevolen een standaard kandidaat-beheersjabloon op te zetten, die kan worden aangepast met kandidaat-informatie zoals deze wordt geïdentificeerd.

Niet alle organisaties hebben een geavanceerde labomgeving die de productieomgeving gemakkelijk kan nabootsen. Sommige organisaties slaan lab testen af vanwege de kosten en de mogelijkheid om een nieuwe versie in het netwerk te testen zonder grote zakelijke impact. Desondanks worden organisaties met hoge beschikbaarheid aangemoedigd om een lab te bouwen dat het productienetwerk na elkaar maakt en om een test/validatieproces te ontwikkelen om hoge testdekking voor nieuwe Cisco IOS versies te verzekeren. Een organisatie zou ongeveer zes maanden moeten toestaan om het lab te bouwen. Gedurende deze tijd moet de organisatie werken aan het opstellen van specifieke testplannen en -processen om ervoor te zorgen dat het lab ten volle wordt gebruikt. Voor Cisco IOS betekent dit het maken van specifieke Cisco IOS testplannen voor elk vereist software-spoor. Deze processen zijn in grotere organisaties van cruciaal belang, omdat veel laboratoria ongebruikt blijven voor nieuwe producten- en softwareintroducties.

De volgende secties beschrijven kortstondig kandidaat beheer en test/validatie hulpmiddelen om voor Cisco IOS selectie en validatie te gebruiken.

Kandidaat-beheertools

Opmerking: Als u de meeste onderstaande gereedschappen wilt gebruiken, moet u een geregistreerde gebruiker zijn en moet u aangemeld zijn.

- Releaseopmerkingen — Bevat informatie over de hardware, module en ondersteuning van een release. Releaseopmerkingen moeten tijdens het beheer van de kandidaat worden herzien om ervoor te zorgen dat alle vereiste hardware- en softwareondersteuning in de mogelijke release aanwezig is en om eventuele migratiekwesties te begrijpen, waaronder verschillende standaards- of upgradevereisten.

Tools voor testen en valideren

Testen- en valideringsgereedschappen worden gebruikt voor het testen en valideren van netwerkoplossingen, inclusief nieuwe hardware, software en toepassingen.

- **Traffic Generators** - genereren multi-protocol verkeersstromen en ruwe pakketsnelheden die worden gebruikt om de snelheid in een bepaalde link te modelleren met behulp van specifieke protocollen. De gebruikers kunnen de bron-, bestemming MAC- en socket getallen specificeren, Deze waarden kunnen bij gespecificeerde stappen worden verhoogd of worden ingesteld om statisch/vast te zijn of in willekeurige stappen. Verkeersgeneratoren kunnen de pakketten genereren voor de volgende protocollen:IPInternetwork Packet Exchange (IPX)DECnetappelXerox Network Systems (XNS)Internet Control Message Protocol (ICMP)Internet Group Management Protocol (IGMP)Connected Network Service (CLNS)User Datagram Protocol (UDP)Virtual Integrated Network Service (VINES)Data Link PacketEr zijn tools beschikbaar bij Agilent - en Spirent Communications .

- **Packet Counter/Capture/Decoder (Sniffer)** - Hiermee kan de klant pakketten op alle pakket- en datalink-lagen selectief opnemen en decoderen. Het gereedschap heeft de mogelijkheid om de gebruiker de filters te laten specificeren, wat het opnemen van alleen gespecificeerde protocolgegevens toestaat. Filters staan de gebruiker verder toe om het opnemen van de pakketten die een bepaald IP adres, poortnummer of MAC-adres overeenkomen te specificeren. Er zijn tools beschikbaar bij Sniffertechnologieën .
- **Netwerksimulator/**simulator stelt de klant in staat de routingtabellen van specifieke routers te bevolken, op basis van de vereisten van het productienetwerk. Ondersteunt de generatie van IP Routing Information Protocol (RIP), OSPF, Intermediate System-to-Intermediate System (IS-IS), Interior Gateway Routing Protocol (IGRP), Enhanced IGRP (DHCP) en Border Gateway Protocol (BGP) routers. Tools zijn beschikbaar bij PacketStorm Communications en Spirent Communications.
- **Session Emulation**-Generate shding venster voor multiprotocol verkeersstromen en zijn in staat om multiprotocol verkeersstromen over het testnetwerk naar het ontvangende apparaat te verzenden. Het ontvangende apparaat weerklinkt de pakketten terug naar de bron. Het bronapparaat verifieert het aantal pakketten die worden verzonden, ontvangen, uit reeks pakketten, en foutenpakketten. Het gereedschap biedt ook de flexibiliteit om de vensterparameters te definiëren in Transmission Control Protocol (TCP), en zo de client/server traffic sessies in het labnetwerk nauw na te bootsen. De tools zijn beschikbaar bij Empirix .
- **Grote netwerkemulators** - Help bij het testen van de schaalbaarheid van grotere omgevingen. Deze hulpmiddelen zijn in staat om controletype verkeer in een laboratoriumtopologie te creëren en gemakkelijk te injecteren om een productieomgeving nauwer te nabootsen. De mogelijkheden omvatten routeinjectors, protocol buren en Layer 2 protocol buren. Er zijn tools beschikbaar bij Agilent - en Spirent Communications .
- **WAN Simulators** - ideaal voor het testen van het verkeer van de bedrijfstoepassingen waar de bandbreedte en de vertraging potentieel een probleem zijn. Deze tools staan organisaties toe om een toepassing plaatselijk te testen met de geschatte vertraging en bandbreedte om te zien hoe de toepassing via WAN functioneert. Deze instrumenten worden vaak gebruikt voor de ontwikkeling van toepassingen en voor het opstellen van testtypes voor toepassingen binnen ondernemersorganisaties. Adtech, een divisie van Spirent Communications en Shunra bieden WAN-simulatietools.

Kandidaatbeheer

Candidate management is het proces van het identificeren van vereisten voor softwareversie en mogelijke risico's voor de specifieke hardware en enabled-functiesets. Aanbevolen wordt dat een organisatie vier tot acht uur lang naar behoren onderzoek doet naar softwarevereisten, release notes, softwaredefecten en potentiële risico's voordat een release wordt geprojecteerd. Hieronder wordt de basis gelegd voor het beheer van de kandidaat-lidstaten:

- Identificeer softwarekandidaten via Cisco Connection Online (CCO)-tools.
- Looptijd van de software voor risicoanalyse, nieuwe functie of ondersteuning van de code.
- Bekende softwareproblemen, problemen en vereisten tijdens de hele levenscyclus identificeren en bijhouden.
- Identificeer standaardconfiguratiegedrag van het geselecteerde beeld.
- Handhaving van back-out- en uitloopkandidaten voor mogelijke kandidaat-veranderingen.
- Geur-schrobben.

- Cisco ondersteuning voor geavanceerde services

Het identificeren van softwarekandidaten werd complexer met het toenemende aantal Cisco-producties en -softwaretreinen. CCO heeft nu verschillende tools, waaronder Cisco IOS-upgradeplanner, software-zoektool, software-hardware compatibiliteitsmatrix en het product-upgradegereedschap dat organisaties kan helpen potentiële release kandidaten te identificeren. Deze gereedschappen zijn te vinden op http://www.cisco.com/cisco/software/navigator.html.

Vervolgens moet u het risico van de potentiële kandidaat-software analyseren. Dit is het proces van inzicht waar de software momenteel op de looptijdcurve staat en vervolgens de vereisten voor de invoering afwentelen met het potentiële risico van de uitgiftandidaat. Als een organisatie bijvoorbeeld de vroege implementatiesoftware (ED) in een kritieke omgeving met hoge beschikbaarheid wil brengen, moet de bijbehorende risico- en middeleneis voor succesvolle certificering in overweging worden genomen. Een organisatie moet ten minste middelen voor softwarebeheer toevoegen voor situaties met een hoger risico om succes te garanderen. Anderzijds, als er een algemene versie van de implementatie (GD) beschikbaar is die voldoet aan de behoeften van een organisatie, dan zijn er minder middelen voor softwarebeheer nodig.

Wanneer mogelijke lozingen en risico's worden geïdentificeerd, moet u een bug draaien om te bepalen of er al dan niet geïdentificeerde catastrofale insecten bestaan die certificering mogelijk verhinderen. Cisco's Bug Watcher, Bug Navigator en Bug Watcher Agents kunnen mogelijke problemen helpen onderkennen en moeten tijdens de gehele levenscyclus van de software worden gebruikt om mogelijke problemen op het gebied van beveiliging of defect te identificeren.

Een nieuwe softwarekandidaat zou ook moeten worden herzien voor mogelijk standaardconfiguratiegedrag. Dit kan worden bereikt door de releaseopmerkingen voor het nieuwe softwarebeeld te bekijken en door configuratieverschillen te bekijken met het potentiële beeld dat op de aangewezen platforms is geladen. Kandidaatbeheer kan ook de identificatie van back-out-versies of go-to-versies omvatten indien de gekozen versie op een bepaald moment in het proces niet aan certificeringscriteria voldoet. Door te kijken naar insecten gerelateerd aan eigenschappen voor een bepaald spoor kan een organisatie potentiële kandidaten voor certificering onderhouden.

Cisco Advanced Services is ook een uitstekend gereedschap voor kandidaatbeheer. Deze groep kan verder inzicht bieden in het ontwikkelingsproces en samenwerking tussen een groot aantal experts uit de industrie in veel verschillende omgevingen op de verticale markt. Meestal zijn de beste bug-scrupules of kandidaat-beheerfuncties binnen de ondersteuning van Cisco, dankzij de expertise en zichtbaarheid in productiesoftware versies die op andere organisaties actief zijn.

Testen en valideren

Testen en validering is een cruciaal aspect van de beste beheerpraktijken en een hoge beschikbaarheid van netwerken. Correct testen van het lab kan de productie aanzienlijk beperken, helpt het ondersteunend personeel van het netwerk te trainen en ondersteunt het stroomlijnen van de processen voor netwerkimplementatie. Om effectief te zijn moet de organisatie de nodige middelen toewijzen om de geschikte labomgeving op te bouwen en te onderhouden, de nodige middelen inzetten om de juiste tests uit te voeren en een aanbevolen testmethode gebruiken die meetverzameling omvat. Zonder een van deze gebieden kan een test- en valideringsproces niet aan de verwachtingen van een organisatie voldoen.

De meeste bedrijfsorganisaties hebben niet de aanbevolen testlabomgeving. Om deze reden hebben veel organisaties oplossingen onjuist ingezet, problemen met netwerkverandering ervaren of softwareproblemen ervaren die in een labomgeving konden worden geïsoleerd. In sommige omgevingen is dit aanvaardbaar, omdat de kosten van de downtime de kosten van een

geavanceerde labomgeving niet compenseren. In veel organisaties kan downtime echter niet worden getolereerd. Deze organisaties worden sterk aangespoord om de aanbevolen testlaboratoria, testtypes en testmethodologieën te ontwikkelen om de kwaliteit van het productienetwerk te verbeteren.

Het Test Lab en de Environment

Het laboratorium moet een geïsoleerd gebied zijn met voldoende ruimte voor bureaus, werkbanken, testapparatuur en behuizingen of racks van de apparatuur. De meeste grote organisaties zullen tussen vier en tien racks apparatuur nodig hebben om de productieomgeving na te bootsen. Er wordt enige fysieke beveiliging aanbevolen om te helpen een testomgeving te onderhouden terwijl de tests bezig zijn. Dit helpt te voorkomen dat een laboratoriumtest wordt onderbroken als gevolg van andere lab-prioriteiten waaronder het lenen van hardware, training of implementatierepetities. Logische veiligheid wordt ook aanbevolen om te voorkomen dat nobele routes het productienetwerk binnendringen of dat ongewenst verkeer het lab verlaat. Dit kan worden gedaan met het routeren van filters en uitgebreide toegangslijsten op een lab gateway router. Connectiviteit met het productienetwerk is behulpzaam voor softwaredownloads en toegang tot het labnetwerk vanuit de productieomgeving.

De laboratoriumtopologie zou de productieomgeving voor om het even welke specifieke testplannen moeten kunnen nabootsen. Het reproduceren van hardware, netwerktopologie, en functieknoppen wordt aanbevolen. Het is uiteraard vrijwel onmogelijk de werkelijke topologie te reproduceren, maar wat kan worden gedaan is de netwerkhiërarchie en de interactie tussen de productieapparaten te reproduceren. Dit is belangrijk voor protocol of eigenschap interactie tussen meerdere apparaten. Sommige testtopologieën zullen verschillen op basis van de vereisten van de softwaretest. De rand van Cisco IOS van WAN zou bijvoorbeeld geen LAN type apparaten of het testen moeten vereisen en kan slechts WAN randrouters en WAN distributirouters nodig hebben. De sleutel is om software functionaliteit na te bootsen zonder productie te kopiëren. In sommige gevallen kunnen de gereedschappen zelfs worden gebruikt om grootschalig gedrag na te bootsen, zoals getallen van de protocolburen en tabellen voor het routeren.

Er zijn ook tools nodig om te helpen bij bepaalde testtypen door het vermogen te verbeteren om de productieomgeving te nabootsen en testgegevens te verzamelen. Tot de hulpmiddelen die de productie helpen nabootsen behoren verkeersverzamelaars, verkeersgenerators, en WAN simulatieapparaten. Smartbits is een goed voorbeeld van een apparaat dat netwerkverkeer kan verzamelen en opnieuw afspelen of grote verkeersvolumes kan genereren. Een organisatie kan ook profiteren van apparaten die kunnen helpen gegevens te verzamelen, zoals protocolanalysatoren.

Het lab heeft ook enig management nodig. Veel grotere organisaties hebben een fulltime lab manager die de verantwoordelijkheid heeft voor het beheer van het labnetwerk. Andere organisaties gebruiken bestaande architectuur en technische teams voor laboratoriumvalidatie. De taken van het Lab beheer omvatten het bestellen van laboratoriumapparatuur en het volgen van activa, het bedraden, fysiek ruimtebeheer, het definiëren van lab regels en richting, laboratoriumplanning, het opzetten van laboratoriumtopologieën, het schrijven van testplannen, het uitvoeren van laboratoriumtesten, en het beheren van potentiële geïdentificeerde kwesties.

Testtypen

Over het geheel genomen zijn er veel verschillende soorten tests die kunnen worden uitgevoerd. Alvorens een volledig testlaboratorium en testplan te bouwen dat alles in een verscheidenheid van configuraties kan testen, zou een organisatie de verschillende soorten van testen, de bedoeling van de testen moeten begrijpen en of de engineering, technische marketing of klantenverdediging

van Cisco verantwoordelijk zouden of kunnen zijn voor een aantal van de verschillende testen. In de testplannen van de klanten worden over het algemeen de meer blootgestelde testtypes behandeld. De volgende tabel helpt de verschillende testtypen te begrijpen, wanneer de tests moeten worden uitgevoerd, en de verantwoordelijke partijen.

Van de onderstaande testen is een juiste test van de specifieke functieset van een organisatie, topologie en toepassingsmix normaal de meest waardevolle. Het is belangrijk om te weten dat Cisco volledige optie- en regressietests uitvoert, echter kan Cisco het toepassingsprofiel van uw organisatie niet met uw specifieke combinatie van topologie, hardware, en geconfigureerde eigenschappen testen. In feite is het ondoenlijk om het volledige scala van eigenschappen, hardware, modules, en topologieën te testen. Daarnaast kan Cisco de interoperabiliteit niet testen met apparatuur van derden. Cisco raadt organisaties aan de nauwkeurige combinatie van hardware, modules, functies en topologie te testen die in hun omgeving gevonden wordt. Deze test moet in een lab worden uitgevoerd, met een samengevouwen topologie die de productieomgeving van uw organisatie representeert met andere ondersteunende testtypes zoals prestaties, interoperabiliteit, stroomuitval en branding-in.

Test	Overzicht van tests	Testverantwoordelijkheid
Functionaliteit en functie	Bepaalt als de basiseigenschappen van Cisco IOS en de hardwaremodules van Cisco zoals geadverteerd. Functie- of modulefunctionaliteit en opties voor configuratie van de functies moeten worden getest. Configuratie-verwijdering en -toevoeging moeten worden getest. De basisuitvaltests en de brandtest zijn inbegrepen.	Cisco-apparaattesten
regressie	Bepaalt als de functie of module in combinatie met andere modules en functies werkt en als de Cisco IOS versie in combinatie met andere Cisco IOS versies op de gedefinieerde	Cisco-regressietest

	functies werkt. Omvat wat verbranding en stroomuitval testen.	
Basisprestaties van het apparaat	Bepaal de basisprestaties van de functie of module om te bepalen of de Cisco IOS optie of hardwaremodules aan minimumvereisten onder lading voldoen.	Cisco-apparaattesten
Topologie/functies/hardwarecombinatie	Bepaalt of de eigenschappen en modules zoals verwacht in een specifieke topologie en module/eigenschap/hardwarecombinatie functioneren. Deze tests moeten protocolverificatie, functietoetsing, controle van de opdracht, inbraaktests en outage-tests omvatten.	Cisco test standaard geadverteerde topologieën in laboratoria zoals Enterprise Solutions Engineering (ESE) en netwerkworked solutions Integratietechniek (NSITE). Hoge beschikbaarheid klanten moeten eigenschappen/module/topologiecombinaties zoals vereist testen, vooral met vroege adoptiesoftware en niet-standaard topologieën.
Afstand (Wat-indien)	Omvat gemeenschappelijke outage types of gedragingen die kunnen voorkomen in een specifieke eigenschap/module/topologie en potentiële functionaliteit impact. Uitgangstesten	Cisco is verantwoordelijk voor basisuitvaltests. Klanten zijn uiteindelijk verantwoordelijk voor problemen in verband met de uitvalprestaties die verband houden met de schaalbaarheid van

	omvatten het omwisselen van kaarten, het opvullen van koppelingen, het uitvallen van apparaten, het falen van verbindingen en het falen van de kaart.	hun individuele omgeving. Afvaltests moeten indien mogelijk in de laboratoriumomgeving van de klant worden uitgevoerd.
Netwerkprestaties (Wat-als)	Onderzoek de lading van het apparaat met betrekking tot een specifieke eigenschap/hardware/topologie combinatie. De nadruk ligt op de capaciteit en prestaties van het apparaat, zoals CPU, geheugen, buffergebruik, en verbindingsgebruik met betrekking tot een vastgesteld verkeerstype en hulpmiddelvereisten voor protocollen, buren, aantal routes, en andere functies. De test helpt de schaalbaarheid in grotere omgevingen te waarborgen.	Klanten zijn uiteindelijk verantwoordelijk voor het laden en schaalbaar apparaat. De lading en de schaalbaarheidskwesties worden vaak door de verkoop van Cisco of de Geavanceerde services aan de orde gesteld en worden vaak getest met Cisco laboratoria zoals de Klantenbewijs-van-Concept Labs (CPOC).
Bug Fix	Zorgt ervoor dat insecten de vastgestelde defect repareren.	Cisco test bug fixes om te verzekeren dat bug is gerepareerd. Klanten dienen ook te testen om er zeker van te zijn dat de bug die ze hebben ervaren is gerepareerd en dat de bug geen ander

		aspect van de module of functie breekt. Onderhoudsreleases zijn regressie-getest, maar tussentijdse releases zijn gewoonlijk niet.
Netwerkbeheer	Verzoekt Simple Network Management Protocol (SNMP)-beheerfuncties, SNMP MIB-variabele nauwkeurigheid, valondersteuning en systeemondersteuning.	Cisco is verantwoordelijk voor het testen van basisfuncties van SNMP, functionaliteit en MIB variabele nauwkeurigheid. Klanten moeten netwerkbeheerresultaten valideren en uiteindelijk verantwoordelijk zijn voor de beheerstrategie en -methodologie voor nieuwe technologieimplementaties.
Grote netwerkemulatie	De grootschalige netwerkemulatie maakt gebruik van tools zoals de routersimulator van Agilent en de testgereedschapsreeks van Spirent om grotere omgevingen te simuleren. Dit kan protocol buren, frame-relais permanente virtuele Circuit (PVC) tellingen, routingtabellen, cache items en andere bronnen omvatten die normaal vereist zijn in productie	De klanten van Cisco zijn in het algemeen verantwoordelijk voor de aspecten van netwerk simulatie die hun netwerkomgeving reproduceert, die het aantal kan omvatten van het routing protocol buren / nabijheden en bijbehorende routing tabel en andere bronnen die in productie zijn.

	en die niet standaard in het lab zijn.	
Interoperabiliteit	Tests alle aspecten met betrekking tot connectiviteit op netwerkapparatuur van derden, vooral als protocol of signaleringsinteroperabiliteit vereist is.	De klanten van Cisco zijn in het algemeen verantwoordelijk voor alle aspecten van interoperabiliteitstesten.
ingebrand	Onderzoek routerresources in de tijd. Voor ingebouwde tests is het meestal nodig dat een apparaat onder een bepaalde lading valt, terwijl er onderzoek wordt gedaan naar het gebruik van hulpbronnen, inclusief geheugen, CPU en buffers in de loop der tijd.	Cisco voert fundamentele inbrandtests uit. Het testen van klanten wordt aanbevolen met betrekking tot unieke topologieën, apparaten en functiescombinaties.

Testmethode

Zodra een organisatie weet wat zij testen, moet een methodologie voor het testproces worden ontwikkeld. Het doel van een testmethode met beste praktijken is te helpen waarborgen dat de overeengekomen tests alomvattend, goed gedocumenteerd, gemakkelijk reproduceerbaar en waardevol zijn voor het vinden van potentiële productieproblemen. Documentatie en het opnieuw genereren van laboratoriumscenario's is vooral belangrijk voor het testen van latere versies of voor het testen van bug fixes in de laboratoriumomgeving. De stappen van een testmethodologie worden hierna weergegeven. Sommige teststappen kunnen ook tegelijkertijd worden uitgevoerd.

1. Maak een testtopologie die de geteste productieomgeving simuleert. Een WAN Edge-testomgeving kan alleen een paar kernrouters en één randrouter omvatten, terwijl een LAN-test ook meer apparaten kan omvatten die de omgeving het best kunnen weergeven.
2. Configureer functies die de productieomgeving simuleren. De configuratie van de laboratoriumapparatuur dient nauw te aansluiten bij de verwachte hardware- en softwareconfiguraties van de productieapparatuur.
3. Schrijf een testplan, definieer testen en doelen, documenteer de topologie en definieer functionele testen. Tests omvatten basisprotocol validatie, tonen bevelvalidatie, outage testen, en brandwonden testen. In de volgende tabel wordt een voorbeeld van een specifieke

test in een testplan gegeven.

4. Verifieer routing en protocolfunctionaliteit. Verwacht document of baseline **toont** opdrachtresultaten. Protocols moeten zowel Layer 2-protocollen zoals ATM, Frame Relay, Cisco Discovery Protocol (CDP), Ethernet en Spanning-Tree evenals Layer 3-protocollen zoals IP, IPX en multicast omvatten.
5. Functionaliteit valideren. Verwacht document of baseline **toont** opdrachtresultaten. De functies kunnen mondiale configuratieopdrachten en alle belangrijke functies zoals verificatie, autorisatie en accounting (AAA) omvatten.
6. Simuleer de lading, wat in de productieomgeving verwacht zou worden. De simulatie van de lading kan worden uitgevoerd met verkeersophalers / generatoren. Verifieer de verwachte toepassingsvariabelen van het netwerkapparaat, inclusief CPU, geheugen, buffergebruik en interfacestatistieken met een onderzoek naar pakketverlies. Verwacht document of baseline **toont** opdrachtresultaten.
7. Uitvaltests uitvoeren waarbij van de machine en de software wordt verwacht dat hij met onderbelasting omgaan of onderbelasting voorkomen. Bijvoorbeeld: kaartverwijdering, link flapping, routeflapping, en uitzending stormen. Zorg ervoor dat de juiste SNMP-traps worden gegenereerd op basis van de functies die binnen het netwerk worden gebruikt.
8. De resultaten van de documenttest en de metingen van de apparatuur als tests moeten worden herhaald.

Testnaam	**Hot Standby Router Protocol (HSRP)-failover**
Testconfiguratie	Pas de lading toe op de primaire gateway interface. Het verkeer moet 20% zijn naar de poort vanuit het perspectief van de gebruikersstations en 60% naar het perspectief van de gebruikersstations. Vergroot ook het verkeer naar een hogere lading.
Teststappen	Monitoren STP en HSRP via opdrachten **tonen**. Schakel de primaire verbinding van de gateway uit en herstel vervolgens de verbinding nadat de informatie is verzameld.
Verwacht metingen	CPU tijdens failover. Toon de interface voor, tijdens, en na voor de primaire en secundaire gateway. HSRP tonen voor, tijdens en na.
Verwachte resultaten	Primaire gateway faalt binnen twee seconden op de andere routergateway. **Laat** opdrachten de verandering goed weergeven. Uitschakelen naar de primaire gateway doet zich voor wanneer de connectiviteit wordt hersteld.
Feitelijke resultaten	
Doorgeven of falen	
Wijzigingen	

vereist om passeren te bereiken	

Apparaatmetingen

Voer tijdens de testfase de volgende metingen uit en documenteer deze om ervoor te zorgen dat de machine correct functioneert:

- Geheugengebruik
- CPU-ladingen
- buffergebruik
- Interfacestatistieken
- Routertabellen
- Specifieke zuivering

De informatie voor metingen varieert afhankelijk van de uitgevoerde specifieke test. Afhankelijk van de specifieke kwesties die aan de orde worden gesteld, kan er ook aanvullende informatie voor meting zijn.

Voor elke toepassing die wordt getest, meet parameters om te waarborgen dat er geen nadelige invloed op de prestaties van de desbetreffende toepassing is. Dit wordt voltooid door gebruik te maken van een prestatiekarakter die kan worden gebruikt om prestaties vóór en na plaatsing te vergelijken. Voorbeelden van meettests voor de toepassing zijn:

- De gemiddelde tijd die nodig is om op een netwerk te loggen.
- De gemiddelde tijd die nodig is om een groep bestanden te kopiëren (NFS) van Network File System.
- De gemiddelde tijd die nodig is om een applicatie te starten, wordt gegenereerd met het eerste scherm.
- Andere toepassingsspecifieke parameters.

Implementatie - Snelle en succesvolle Cisco IOS-implementaties

Een goed gedefinieerd implementatieproces maakt een organisatie mogelijk om nieuwe Cisco IOS-versies efficiënt in te voeren.

De uitvoeringsfase omvat het proefproces en het uitvoeringsproces. Het proefproces zorgt ervoor dat de Cisco IOS versie in de omgeving succesvol zal zijn en het implementatieproces snelle en succesvolle grotere Cisco IOS-implementaties mogelijk maakt.

Strategie en tools voor Cisco IOS-implementaties

De strategie voor Cisco IOS-implementaties is het uitvoeren van definitieve certificering via een proefproces en snelle implementatie met behulp van upgradetools en een duidelijk omschreven implementatieproces.

Alvorens een netwerkproefproces op te zetten, bouwen veel organisaties algemene

proefrichtlijnen op. De modelrichtsnoeren moeten verwachtingen bevatten voor alle piloten, zoals succescriteria, aanvaardbare proeflocaties, documentatie van piloten, verwachtingen van eigenaars van piloten, eisen voor de melding van gebruikers en verwachte duur van de piloot. Een functioneel team van ingenieurs, implementaties en operaties is doorgaans betrokken bij de ontwikkeling van algemene proefrichtlijnen en een proefproces. Wanneer het proefproces eenmaal tot stand is gebracht, kunnen afzonderlijke uitvoeringsgroepen doorgaans succesvolle piloten uitvoeren met gebruikmaking van de vastgestelde beste praktijkmethoden.

Zodra een nieuwe softwareversie voor plaatsing en definitieve certificering is goedgekeurd, moet de organisatie beginnen met het plannen van de Cisco IOS-upgrade. De planning begint met het identificeren van nieuwe beeldvereisten waaronder platform, geheugen, flitser en configuratie. De architectuur en technische groepen definiëren normaal de nieuwe vereisten voor softwareafbeelding in de kandidaat-beheerfase van de Cisco IOS-beheercyclus. Zodra de eisen zijn vastgesteld, moet elk hulpmiddel door de uitvoeringsgroep worden gevalideerd en eventueel aangepast. De module CiscoWorks2000 van het Afbeeldingsbeheer van de software (SWIM) kan ook de valideringsstap uitvoeren door Cisco IOS-vereisten tegen apparaatinventaris te valideren. Wanneer alle apparaten gevalideerd en of bijgewerkt zijn tot de juiste nieuwe beeldstandaarden, kan de implementatiegroep een langzaam-start implementatieproces starten met behulp van de CiscoWorks2000 SWIM-module als software-implementatiegereedschap.

Nadat het nieuwe beeld met succes een aantal keren is ingezet, kan de organisatie een snelle plaatsing beginnen met gebruik van CiscoWorks SWIM.

Cisco IOS contentbeheer

De manager van de Uitrustingscapaciteit van CiscoWorks2000 van het Resourcegids Manager (RME) vereenvoudigt zeer het versiebeheer van Cisco routers en switches door op web-gebaseerde rapportagetools die Cisco IOS apparaten op basis van softwareversie, platform en apparaatnaam rapporteren en sorteren.

Cisco IOS SWIM

CiscoWorks2000 SWIM kan helpen om de foutgevoelige complexiteit van het upgradeproces te verminderen. Ingebouwde links naar CCO correleren de online informatie van Cisco over softwarepatches met Cisco IOS en Catalyst software die in het netwerk wordt geïnstalleerd, en markeren verwante technologie notities. Nieuwe planningsgereedschappen vinden systeemvereisten en verzenden meldingen als hardware-upgrades (Opstarten-ROM, Flash RAM) nodig zijn ter ondersteuning van voorgestelde software-beeldupdates.

Voordat een update wordt gestart, worden de vereisten voor een nieuw beeld gevalideerd tegen de inventarisgegevens van de doelswitch of router om een succesvolle upgrade te waarborgen. Wanneer meerdere apparaten worden bijgewerkt, synchroniseert SWIM downloadtaken en stelt de gebruiker in staat om de voortgang van de taak te controleren. Geplande banen worden gecontroleerd door een automatische procedure, waardoor managers de activiteiten van een technicus kunnen autoriseren voordat ze elke upgrade-taak starten. RME 3.3 biedt de mogelijkheid om softwareupgrades te analyseren voor Cisco IGX-, BPX- en MGX-platforms, waardoor de tijd die nodig is om het effect van een softwareupgrade te bepalen, aanzienlijk vereenvoudigd en verkort.

proefproces

Om de potentiële blootstelling tot een minimum te beperken en eventuele resterende

productieproblemen op een veiliger wijze op te vangen, wordt een softwarepiloot aanbevolen. Piloten zijn over het algemeen belangrijker voor nieuwe technologieimplementaties, hoe veel nieuwe softwareimplementaties zullen worden gekoppeld aan nieuwe services, functies of hardware, waar een piloot kritischer is. In het individuele proefplan moeten proefselectie, proefduur en meting worden overwogen. Proefselectie is het proces om te bepalen wanneer en waar een piloot moet worden uitgevoerd. Proefmetingen zijn het proces van het verzamelen van de vereiste gegevens om succes en falen of potentiële problemen te identificeren.

Proefselectie identificeert waar en hoe een piloot zal worden voltooid. Een piloot kan met één apparaat starten in een gebied met lage botsing en zich uitbreiden tot meerdere apparaten in een gebied met een hogere botsing. Enkele overwegingen voor proefselectie waar de impact kan worden beperkt zijn:

- Geïnstalleerd in een gebied van het netwerk, veerkrachtig tegen één apparaat door redundantie.
- In een gebied van het netwerk met een minimaal aantal gebruikers achter het geselecteerde apparaat die met wat mogelijke impact op de productie kunnen omgaan.
- Overweeg de piloot langs architectuurlijnen te scheiden. Bijvoorbeeld, besturen het in de toegang, distributie, en/of kernlagen van het netwerk.

De duur van deze proef moet gebaseerd zijn op de tijd die nodig is om alle voorzieningen voldoende te testen en te evalueren. Hieronder vallen zowel de brandwonden als het netwerk bij normale verkeersladingen. De duur is ook afhankelijk van de stap in de codeupgrade en het gebied van het netwerk waar Cisco IOS actief is. Als Cisco IOS een nieuwe belangrijke release is, heeft u de voorkeur voor een langere proefperiode. Wanneer de upgrade een onderhoudsrelease is met minimale nieuwe functies, is een kortere proefperiode voldoende.

Tijdens de proeffase is het belangrijk de resultaten op dezelfde wijze te controleren en te documenteren als de eerste tests. Dit kan gebruikersenquêtes, het verzamelen van proefgegevens, het verzamelen van problemen en de criteria voor succes/falen omvatten. Individuen moeten rechtstreeks verantwoordelijk zijn voor het volgen en bewaken van de voortgang van de proefprojecten om ervoor te zorgen dat alle kwesties worden geïdentificeerd en dat de gebruikers en diensten die bij de proef betrokken zijn, tevreden zijn met de proefresultaten. De meeste organisaties zullen een release certificeren als deze succesvol is in een piloot- of productieomgeving. Deze stap is in sommige omgevingen een kritieke tekortkoming als gevolg van een waargenomen succes wanneer geen meet- of succescriteria zijn vastgesteld of gedocumenteerd.

Uitvoering

Nadat de proeffase binnen het productienetwerk is voltooid, begin de Cisco IOS implementatiefase. De implementatiefase omvat verschillende stappen om ervoor te zorgen dat de software een succesvolle en efficiëntere implementatie waarborgt, waaronder een trage start, definitieve certificering, upgradevoorbereiding, upgradeautomatisering en definitieve validatie.

Langdurige start van de implementatie is het proces om langzaam een nieuw geteste release uit te voeren om ervoor te zorgen dat het beeld volledig is blootgesteld aan de productieomgeving vóór de definitieve certificering en volledige schaalomzetting. Sommige organisaties kunnen beginnen met één toestel en één dag blootstelling voordat ze de volgende dag overgaan op twee apparaatupgrades en wellicht een paar dagen later. Wanneer ongeveer tien apparaten in productie zijn geplaatst, kan de organisatie tot één tot twee weken vóór de definitieve certificering van de specifieke Cisco IOS-versie wachten. Na de definitieve certificering kan de organisatie de geïdentificeerde versie sneller met een veel hoger betrouwbaarheidsniveau implementeren.

Na het proces van langzaam starten moeten alle apparaten die voor upgrade geïdentificeerd zijn, worden beoordeeld en gevalideerd met behulp van de apparaatinventaris en een matrix van de minimale Cisco IOS-normen voor bootstrap, DRAM en flitser om ervoor te zorgen dat aan de vereisten wordt voldaan. De gegevens kunnen door in-house gereedschappen, SNMP-tools van derden of door het gebruik van CiscoWorks2000 RME worden aangeschaft. De CiscoWorks2000 SWIM herziet of inspecteert deze variabelen vóór de implementatie. Het is echter altijd een goed idee om te weten wat er te verwachten is tijdens de implementatiepogingen.

Als meer dan honderd soortgelijke apparaten voor upgrades zijn gepland, wordt sterk aanbevolen een geautomatiseerde methode te gebruiken. Automatisering heeft aangetoond de upgradeefficiëntie te verbeteren en het percentage van de successen van de upgrade van het apparaat tijdens grote implementaties te verbeteren, op basis van een interne upgrade van 1000-apparaten met en zonder SWIM. Cisco raadt aan om CiscoWorks 2000 SWIM te gebruiken voor grote implementaties vanwege de mate van verificatie die tijdens de upgrade wordt uitgevoerd. SWIM zal zelfs een back-up maken van een Cisco IOS versie als een probleem wordt gedetecteerd. SWIM functioneert door upgradebanen te maken en te plannen, waar een taak met de apparaten is ingesteld, gewenste upgradeafbeeldingen en uitvoertijd van de taak. Elke taak moet twaalf of minder apparaatupgrades bevatten en maximaal twaalf banen kunnen tegelijkertijd worden uitgevoerd. SWIM verifieert ook dat de geplande Cisco IOS upgrade-versie met succes in werking is na de upgrade. Aanbevolen wordt om voor elke upgrade ongeveer twintig minuten toe te staan (inclusief verificatie). Met deze formule kan een organisatie 36 apparaten per uur upgraden. Cisco raadt ook aan om maximaal honderd apparaten per avond te verbeteren om potentiële blootstelling aan problemen te beperken.

Na een geautomatiseerde upgrade moet enige validatie worden uitgevoerd om succes te garanderen. Het SWIM-gereedschap van CiscoWorks2000 kan aangepaste scripts uitvoeren na de upgrade om verdere succesverificatie uit te voeren. Verificatie omvat het valideren dat de router het juiste aantal routes heeft, het verzekeren dat de logische/fysieke interfaces omhoog en actief zijn, of het valideren dat het apparaat toegankelijk is. De volgende steekproefcontrolelijst kan het succes van een Cisco IOS plaatsing volledig valideren:

- Heeft het apparaat goed opnieuw geladen?
- Is het apparaat bereikbaar via de NMS-platforms (Network Management System)?
- Zijn de verwachte interfaces op het apparaat actief?
- Heeft het apparaat de juiste routingprotocol nabijheid?
- Is de routingtabel bevolkt?
- Is het apparaat het verkeer juist passeren?

Operations - beheer van de hoge beschikbaarheid van Cisco IOS-implementatie

Hoge beschikbaarheid van best practice-bewerkingen van de Cisco IOS-omgeving helpen de netwerkcomplexiteit te verminderen, de tijd voor probleemoplossing te verbeteren en de netwerkbeschikbaarheid te verbeteren. Het operationele gedeelte van Cisco IOS beheer omvat strategie, gereedschappen en best practice methodologieën die worden aanbevolen voor het beheer van Cisco IOS.

De beste praktijken voor Cisco IOS operaties omvatten de controle van de softwareversie, Cisco IOS systeembeheer, probleembeheer, configuratie standaardisering, en beschikbaarheidsbeheer. Softwareversiecontrole is het proces van het volgen, valideren en verbeteren van softwareconsistentie binnen de geïdentificeerde softwaresporen. Cisco IOS SLOG-beheer is het

proces van proactief toezicht op en optreden bij systeemmeldingen met hogere prioriteit die door Cisco IOS gegenereerd worden. Problemen beheer is de praktijk om snel en efficiënt kritische probleeminformatie te verzamelen voor softwaregerelateerde kwesties om toekomstige voorvallen te helpen voorkomen. De standaardisering van de configuratie is het proces van het standaardiseren van configuraties om het potentieel voor niet-geteste codes bij de productie te verminderen en netwerkprotocol- en functiegedrag te standaardiseren. Het beheer van de beschikbaarheid is het proces van het verbeteren van beschikbaarheid gebaseerd op parameters, verbeteringsdoelstellingen en verbeteringsprojecten.

Strategieën en tools voor Cisco IOS-bewerkingen

Veel kwaliteitsstrategieën en tools bestaan om Cisco IOS-omgevingen te helpen beheren. De eerste belangrijkste strategie voor Cisco IOS operaties is het milieu zo eenvoudig mogelijk te houden, waarbij variatie in configuratie en Cisco IOS versies zoveel mogelijk wordt voorkomen. Cisco IOS-certificering is al besproken, maar de configuratie-consistentie is een ander belangrijk gebied. De bouwkundige/technische groep dient verantwoordelijk te zijn voor het opstellen van standaarden voor configuratie. De implementatie en de operaties groep hebben dan de verantwoordelijkheid om de standaarden te configureren en onderhouden via Cisco IOS versiecontrole en configuratie standaarden / controle.

De tweede strategie voor Cisco IOS operaties is de mogelijkheid om netwerkfouten te identificeren en snel op te lossen. Netwerkproblemen moeten in het algemeen door de functiegroep worden geïdentificeerd voordat gebruikers ze inroepen. Ook de problemen moeten zo snel mogelijk worden opgelost zonder verdere impact of verandering van het milieu. Een paar belangrijke beste praktijken op dit gebied zijn probleembeheer en Cisco IOS systeembeheer. Een gereedschap om snel te helpen diagnosticeren met Cisco IOS-softwarecrashes is de Cisco O&O-tolk.

De derde strategie is een consistente verbetering. Het primaire proces is het verbeteren van een op kwaliteit gebaseerd programma ter verbetering van de beschikbaarheid. Door een analyse van de oorzaak van alle kwesties, inclusief Cisco IOS verwante kwesties, uit te voeren kan een organisatie testdekking verbeteren, de tijden van probleemoplossing verbeteren en processen verbeteren die outage-impact elimineren of verminderen. De organisatie kan ook naar gemeenschappelijke problemen kijken en processen opbouwen om deze kwesties sneller op te lossen.

Tot de tools voor Cisco IOS operaties behoren voorraadbeheer voor de controle van de softwareversie (CiscoWorks2000 RME), Syslog beheer om Syslog-berichten te beheren, en apparaatconfiguratiemanagers om de consistentie van het apparaat te beheren.

Syrische beheer

De boodschappen van Syslog zijn berichten die door het apparaat naar een verzamelserver worden verstuurd. Deze berichten kunnen fouten zijn (bijvoorbeeld een link die omlaag gaat) of informatie zijn, zoals wanneer iemand er is geweest om een terminal op een apparaat te configureren.

Syrische beheertools registreren en bijhouden de meldingen die door routers en switches worden ontvangen. Sommige gereedschappen hebben filters om ongewenste berichten te verwijderen die de belangrijke kunnen beïnvloeden. Dankzij de instrumenten van het systeem zouden ook de rapportering op basis van de ontvangen berichten moeten kunnen worden gemaakt. De rapportage kan worden weergegeven op basis van de tijdsperiode, het apparaat, het berichttype of de prioriteit van het bericht.

Het populairste gereedschap van de SPRONG voor het beheer van Cisco IOS is CiscoWorks2000 van de Syrische manager van RME. Er zijn ook andere tools beschikbaar, zoals SL4NT, een gedeeld programma van Netal en Private IP van OpenSystems.

CiscoWorks apparaatbeheer

De CiscoWorks2000 Manager van de Configuratie van het apparaat handhaaft een actief archief en verstrekt een makkelijke manier om configuratieveranderingen over meerdere routers en switches van Cisco bij te werken. De configuratiebeheerder controleert het netwerk voor configuratieveranderingen, werkt het archief bij wanneer een verandering wordt gedetecteerd en registreert de veranderingsinformatie aan de Dienst van de Auditing van Verandering. Met een web-gebaseerde gebruikersinterface kunt u het archief doorzoeken naar specifieke configuratieeigenschappen en de inhoud van twee configuratiebestanden vergelijken om een eenvoudige identificatie van verschillen te mogelijk te maken.

Cisco O-tolk

De Cisco-uitvoertolk is een gereedschap dat wordt gebruikt bij het diagnosticeren van softwaregedwongen crashes. Het gereedschap kan u helpen softwaretekortkomingen te identificeren zonder het Cisco Technical Assistance Center (TAC) te bellen, of het kan worden gebruikt als primaire informatie naar de TAC na een softwarecrash. Deze informatie zal in het algemeen bijdragen tot een spoedige oplossing van het probleem, althans wat betreft de vereiste informatie.

Software versie Control

Softwareversiecontrole is het proces van het implementeren van alleen gestandaardiseerde softwareversies en het controleren van het netwerk om software te valideren of mogelijk te wijzigen als gevolg van de niet-versie-conformiteit. In het algemeen wordt de controle van de softwareversie uitgevoerd door middel van een certificeringsproces en een normcontrole. Veel organisaties publiceren versienormen op een centrale webserver. Daarnaast is het implementatiepersoneel opgeleid om te bekijken welke versie in werking is en om de versie bij te werken als deze niet voldoet aan de normen. Sommige organisaties beschikken over een proces van kwaliteitsgate waarbij de secundaire validatie door middel van audits wordt voltooid om ervoor te zorgen dat de norm tijdens de uitvoering wordt nageleefd.

Tijdens de exploitatie is het niet ongebruikelijk om niet-standaardversies in het netwerk te zien, vooral als het netwerk- en operationele personeel groot is. Dit kan zijn veroorzaakt door niet-getraind nieuwer personeel, niet-geconfigureerde laarsopdrachten of niet-gecontroleerde implementaties. Het is altijd een goed idee om softwareversienormen periodiek te valideren met behulp van gereedschappen zoals CiscoWorks 2000 RME die alle apparaten door Cisco IOS versie kunnen sorteren. Als er niet-normen worden vastgesteld, moeten zij onmiddellijk worden aangegeven en moet een kaartje of een ander ticket worden ingelast om de versie naar de vastgestelde standaard te brengen.

Proactief systeembeheer

De inzameling, controle, en analyse van Syslog zijn de processen van het foutbeheer die worden aanbevolen om meer IOS specifieke netwerkproblemen op te lossen die moeilijk of onmogelijk om met andere middelen te identificeren zijn. De inzameling, controle en analyse van het systeem helpen de tijd van de probleemoplossing te verbeteren door vele fouten proactief te identificeren en op te lossen voordat de ernstigste netwerkproblemen worden ervaren, of door gebruikers

worden gemeld. Syslog biedt ook een efficiëntere methode om een grote verscheidenheid aan problemen te verzamelen, in vergelijking met een consistent SNMP-opiniepeiling voor een groot aantal MIB-variabelen. De inzameling, controle, en analyse van het systeem worden verwezenlijkt door het gebruiken van de correcte configuratie, de hulp van Cisco IOS, de hulp van de correlatie, zoals CiscoWorks2000 RME, en/of het beheer van de Syrische gebeurtenis. Het beheer van de Syslog-gebeurtenissen gebeurt door de verzamelde Syslog-gegevens voor geïdentificeerde kritieke berichten te ontkoppelen en vervolgens een waarschuwing of val naar een eventmanager te sturen voor realtime-melding en -oplossing.

Controle op het systeem vereist NMS hulpwerksteun of scripts om te helpen parseren en te rapporteren over Syslog gegevens. Dit omvat de mogelijkheid om Syslog-berichten te sorteren naar datum of tijdsperiode, apparaat, type Syslog-bericht of frequentie van de berichten. In grotere netwerken kunnen tools of scripts worden geïmplementeerd om Syslog-gegevens te parseren en signaleringen of kennisgevingen te verzenden naar beheersystemen voor gebeurtenissen of naar operationele en technische medewerkers. Als er geen signaleringen voor een brede reeks Syrische gegevens worden gebruikt, moet de organisatie ten minste dagelijks hogere prioriteit bieden aan Syslog-gegevens en probleemtickets maken voor mogelijke problemen. Om netwerkproblemen die mogelijk niet door normale bewaking worden gedetecteerd, proactief te detecteren, dienen periodieke review en analyse van historische Syslog-gegevens te worden uitgevoerd om situaties te detecteren die mogelijk geen onmiddellijk probleem aangeven, maar die wel een indicatie van een probleem kunnen zijn voordat de service een effect heeft.

Probleembeheer

Veel klanten ondervinden extra onderbreking door een gebrek aan processen in probleembeheer. Aanvullende downtime kan voorkomen wanneer netwerkbeheerders proberen het probleem snel op te lossen met behulp van een combinatie van opdrachten die de service beïnvloeden of configuratiewijzigingen in plaats van tijd te besteden aan het identificeren van problemen, het verzamelen van informatie en een goed geanalyseerd oplossingspad. Waargenomen gedrag op dit gebied omvat het opnieuw laden van apparaten, of het ontruimen van IP routingtabellen voordat u een probleem en de diepere oorzaak onderzoekt. In sommige gevallen gebeurt dit vanwege de doelstellingen voor probleemoplossing op het eerste niveau. Het doel in alle software-gerelateerde problemen moet zijn om snel de benodigde informatie te verzamelen die nodig is voor de analyse van de oorzaak van de schade voordat de connectiviteit of de service wordt hersteld.

Een probleembeheerproces wordt aanbevolen in grotere omgevingen. Dit proces dient een bepaalde mate van standaardprobleembeschrijvingen te omvatten en geschikte opdrachtverzamelingen te tonen voor de escalatie naar een tweede niveau. De steun op de eerste rij mag nooit worden gebruikt voor het vereffenen van routes of het opnieuw laden van voorzieningen. Optimaal zou de organisatie op het eerste niveau snel informatie moeten verzamelen en naar een tweede niveau moeten escaleren. Door in eerste instantie nog een paar minuten te besteden aan probleemidentificatie of probleembeschrijving is het veel waarschijnlijker dat een ontdekking van de oorzaak van een probleem optreedt, waardoor een tijdelijke oplossing, laboratoriumidentificatie en rapportage van bug mogelijk wordt. Ondersteuning op tweede niveau moet goed worden verwerkt in de soorten informatie die Cisco nodig kan hebben om een probleem te diagnosticeren of een bug-rapport te bestanden. Dit omvat geheugendumps, routinginformatie uitvoer en opdrachtoutput van apparaat.

Standaardisatie voor configuratie

Mondiale normen voor het configureren van apparaten vertegenwoordigen de praktijk van het

handhaven van standaard mondiale configuratieparameters voor alle apparaten en diensten die resulteren in een globale configuratie consistentie voor het gehele bedrijf. Mondiale configuratieopdrachten zijn opdrachten die op het gehele apparaat van toepassing zijn en niet op afzonderlijke poorten, protocollen of interfaces. Mondiale configuratieopdrachten beïnvloeden over het algemeen de toegang tot het apparaat, het algemene gedrag van het apparaat en de veiligheid van het apparaat. In Cisco IOS omvat dit serviceopdrachten, IP-opdrachten, vty opdrachten, console-poortopdrachten, houtkapopdrachten, AAA/TACACS+ opdrachten, SNMP-opdrachten en banner opdrachten. Ook belangrijk in de mondiale standaarden voor apparaatconfiguratie is een geschikte conventie voor het benoemen van apparaten waarmee beheerders het apparaat, het type apparaat en de locatie van het apparaat kunnen identificeren op basis van de naam van het Domain Name System (DNS)-apparaat. Mondiale configuratie consistentie is belangrijk voor de algemene draagbaarheid en betrouwbaarheid van een netwerkomgeving, omdat het de netwerkcomplexiteit helpt verminderen en netwerkondersteuning helpt verbeteren. De moeilijkheid van de ondersteuning wordt vaak ervaren zonder configuratie standaardisatie door incorrect of inconsequent apparaatgedrag, SNMP-toegang en algemene apparaatbeveiliging.

Het handhaven van mondiale normen voor het configureren van apparaten wordt normaal bereikt door een interne engineering of een operationele groep die mondiale configuratieparameters voor gelijksoortige netwerkapparaten creëert en onderhoudt. Het is ook een goede praktijk om een kopie van het globale configuratiebestand in TFTP-telefoongidsen te verstrekken, zodat deze in eerste instantie kunnen worden gedownload naar alle nieuwe voorzieningen. Ook behulpzaam is een web toegankelijk bestand dat het standaard configuratiebestand met een verklaring van elke configuratieparameter bevat. Sommige organisaties configureren mondiaal zelfs op een regelmatige basis soortgelijke apparaten om mondiale configuratie consistentie te verzekeren of apparaten periodiek om de juiste mondiale configuratiestandaarden te controleren. Protocol- en interfacestandaarden vormen de praktijk om standaarden voor interface- en protocolconfiguratie te handhaven.

De consistentie van de protocol en van de interfaceconfiguratie verbetert netwerkbeschikbaarheid door netwerkcomplexiteit te verminderen, verwacht apparaat en protocol gedrag te voorzien en netwerksupportabiliteit te verbeteren. Inconsistentie van de protocol- of interfaceconfiguratie kan resulteren in onverwacht gedrag van het apparaat, problemen bij het routeren van verkeer, problemen met verhoogde connectiviteit en verhoogde reactieve ondersteuningstijd. De interfacestandaarden moeten de beschrijvers van de CDP-interface, de caching-configuratie en andere protocol-specifieke normen omvatten. Protocol-specifieke configuratienormen kunnen omvatten:

- IP-routeringsconfiguratie
- DLSW-configuratie
- Configuratie van toegangslijsten
- ATM-configuratie
- Frame Relay-configuratie
- Spanning-boomconfiguratie
- VLAN-toewijzing en -configuratie
- Virtual Trunking Protocol (VTP)
- HSRP

Opmerking: Het is mogelijk om andere protocol-specifieke configuratiestandaarden te hebben, afhankelijk van wat binnen het netwerk is geconfigureerd.

Een voorbeeld van IP-standaarden kan zijn:

- Subnetgrootte
- IP-adresruimte
- Gebruikte routingprotocol
- Routing protocolconfiguratie

Het in stand houden van protocol- en interfacestandaarden is normaal de verantwoordelijkheid van de technische en implementatiegroepen van het netwerk. De technische groep moet verantwoordelijk zijn voor het identificeren, testen, valideren en documenteren van de normen. De implementatiegroep is dan verantwoordelijk voor het gebruik van de technische documenten of configuratiesjablonen om nieuwe services te leveren. De technische groep moet documentatie creëren over alle aspecten van de vereiste normen om de consistentie te waarborgen. Configuratiescherm moet ook worden gemaakt om de configuratienormen te helpen handhaven. Operationele groepen moeten ook worden opgeleid op het gebied van de normen en moeten in staat zijn niet-standaardconfiguratie-kwesties te identificeren. De consistentie van de configuratie is van groot belang voor de test-, validatie- en certificeringsfase. In feite is het zonder gestandaardiseerde configuratietemplates vrijwel onmogelijk om een CIsco IOS-versie voor een matig groot netwerk adequaat te testen, valideren of certificeren.

Beschikbaarheidsbeheer

Beschikbaarheidsbeheer is het proces van kwaliteitsverbetering door gebruik te maken van netwerkbeschikbaarheid als de kwaliteitsverbetering. Veel organisaties meten nu de beschikbaarheid en het type stroomuitval. Uitgangstypen kunnen hardware, software, link/drager, vermogen/omgeving, ontwerp, of gebruikersfout/proces omvatten. Door stroomstoringen te identificeren en direct na herstel een analyse van de oorzaak uit te voeren kan de organisatie methoden identificeren om de beschikbaarheid te verbeteren. Bijna alle netwerken die een hoge beschikbaarheid hebben bereikt, hebben een kwaliteitsverbeteringsproces.

Bijlage A - Overzicht van Cisco IOS-releases

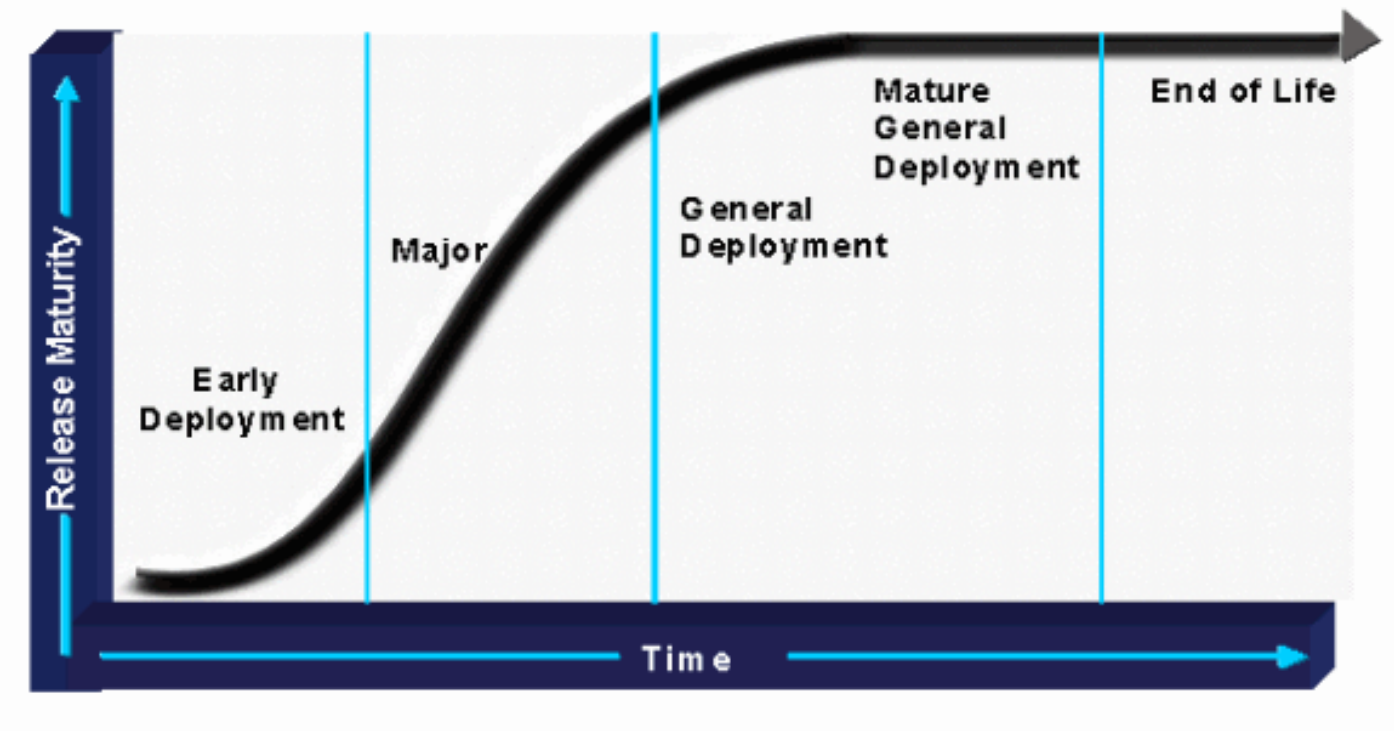
De Cisco IOS-strategie voor softwarerelease is gebaseerd op gezonde softwareontwikkeling, kwaliteitsborging en snelle tijd naar markt, die van fundamenteel belang zijn voor het succes van de netwerken van de klanten van Cisco.

Het proces is gedefinieerd rond vier categorieën releases, die hieronder worden toegelicht:

- Eerste implementaties (ED)
- Belangrijke vrijlating
- Beperkte implementatievrijstelling (LD)
- Algemene implementatievrijstelling (GD)

Cisco maakt en onderhoudt een IOS routekaart die informatie over individuele releases, doelmarkten, migratiepaden, nieuwe functiebeschrijvingen enzovoort bevat.

Het onderstaande figuur illustreert de levenscyclus van de Cisco IOS-softwarerelease:



ED releases

Cisco IOS ED-releases zijn voertuigen die nieuwe ontwikkeling op de markt brengen. Elke onderhoudsrevisie van een ED release bevat niet alleen bug-fixes, maar ook een reeks nieuwe functies, nieuwe platform ondersteuning en algemene verbeteringen van protocollen en de Cisco IOS-infrastructuur. Om de één tot twee jaar, worden de eigenschappen en de platforms van de ED releases geïmporteerd naar de volgende Cisco IOS-release.

Er zijn vier types van de uitstoot van ED, elk met een lichtjes verschillend afgiftemodel en levenscyclusmijlpalen. De ED-releases kan worden geclassificeerd als:

- **Geconsolideerde Technology vroege Deployment (CTED) releases**-het nieuwe Cisco IOS-release model gebruikt de geconsolideerde ED release-trein, ook bekend als de T-trein, om nieuwe functies, nieuwe hardwareplatforms en andere verbeteringen in Cisco IOS te introduceren. Zij worden geconsolideerde technologie genoemd omdat zij de definities van de interne bedrijfseenheden (BU's) en "Line of Business" (LOB) overstijgen. Voorbeelden van geconsolideerde technologische releases zijn Cisco IOS 11.3T, 12.0T en 12.1T.
- **Speciaal technologisch vroege implementaties (STED) releases**-STED-releases hebben dezelfde kenmerken als CTED-releases, behalve dat ze zich richten op een specifieke technologie of markttheater. Ze worden altijd op specifieke platforms uitgebracht en zijn alleen onder supervisie van een Cisco-BU. STED-releases wordt geïdentificeerd met twee letters die aan de belangrijkste release-versie zijn toegevoegd. Voorbeelden van STED-releases zijn Cisco IOS 11.3NA, 11.3MA, 11.3WA en 12.0DA.
- **Speciale releases-van** marktvroeg-**implementaties (SMED)** - de Cisco IOS-**masten-machines** zijn gedifferentieerd van STED's door het feit dat zij zich richten op een specifiek verticaal marktsegment (ISP's, ondernemingen, financiële instellingen, telecombedrijven, enzovoort). SMED's omvatten uitsluitend specifieke eisen inzake technologiekenmerken voor specifieke relevante platforms die door de voorgenomen verticale markt worden gebruikt. Zij kunnen worden onderscheiden van CTED's door het feit dat zij alleen worden gebouwd voor specifieke platforms die relevant zijn voor de verticale markt, terwijl CTED's voor meer platforms zouden worden gebouwd op basis van een breder technologisch vereiste. Cisco

IOS SMED-releases worden geïdentificeerd door één alfabetisch teken dat aan de belangrijkste release-versie is toegevoegd (net zoals de CTED). Voorbeelden van SMED's zijn Cisco IOS 12.0S en 12.1E.
- **Korte, vroege implementaties releases, ook bekend als X releases (XED)**-Cisco IOS XED-releases introduceert nieuwe hardware en technologieën op de markt. Zij verschaffen geen herzieningen van de softwareonderhoudssoftware, noch verschaffen zij reguliere softwaretussentijdse herzieningen. Indien een defect in de XED wordt aangetroffen vóór de convergentie ervan met de CTED, wordt een heropbouw van software gestart en wordt een nummer toegevoegd aan de naam. Bijvoorbeeld, Cisco IOS releases 12.0(2)XB1 en 12.0(2)XB2 zijn voorbeelden van 12.0(2)XB herbouwen.

Belangrijke releases

Belangrijke releases zijn de primaire implementatievoertuigen voor Cisco IOS-softwareproducten. Ze worden beheerd door de Cisco IOS Technology Division en consolideren functies, platforms, functionaliteit, technologie en host-proliferatie uit de vorige ED-releases. Cisco IOS belangrijke releases is gericht op grotere stabiliteit en kwaliteit. Om deze reden accepteren belangrijke releases niet dat alle functies en platforms worden toegevoegd. Elke onderhoudsrevisie biedt alleen foutoplossingen. Cisco IOS-softwarereleases 12.1 en 12.2 zijn bijvoorbeeld belangrijke releases.

Grote releases hebben geplande onderhoudsupdates, genaamd onderhoudsreleases die volledig getest zijn, de meest recente bug-oplossingen bevatten en nieuwe platforms of functies ondersteunen. Het vrijgavenummer van een belangrijke release identificeert de hoofdrelease en het onderhoudsniveau ervan. In Cisco IOS-softwarerelease 12.0(7)is 12.0 het aantal van de belangrijkste release en 7 het onderhoudsniveau. Het volledige release nummer is 12.0(7). Op dezelfde manier is 12.1 een belangrijke release en 12.1(3) is de derde onderhoudsrelease van belangrijke Cisco IOS-softwarerelease 12.1.

Beperkte implementaties (LD) releases

LD is de fase van Cisco IOS rijpheid tussen FCS en algemene implementatie voor hoofdreleases. Cisco IOS ED-releases leeft alleen in de beperkte implementatiefase omdat ze nooit een GD-certificering hebben.

Algemene implementaties (GD) releases

Op een bepaald punt tijdens de levenscyclus van de release zal Cisco een belangrijke release aangeven om klaar te zijn voor de GD-certificering. Alleen een grote release kan een GD-status bereiken. Dit voldoet aan de GD-certificeringsmijlpaal wanneer Cisco ervan overtuigd is dat de release is:

- Bewezen door een uitgebreide marktblootstelling in diverse netwerken.
- Gekwalificeerd door gemeten naar stabiliteit en bug trends.
- Gekwalificeerd door klanttevredenheidsonderzoeken.
- Een vermindering van de genormaliseerde trend van de klant constateerde gebreken in de release ten opzichte van de vorige vier onderhoudsreleases.

Er wordt een klant advocacy GD-certificatie voor kruisfuncties team samengesteld uit TAC-technici, Advanced Engineering Services (AES), System Test Engineering en Cisco IOS Engineering opgericht om elk defect van de release te evalueren. Dit team geeft de definitieve goedkeuring voor de certificering van het GD. Zodra een vrijgave een GD-status heeft, is elke daaropvolgende herziening van de vrijgave ook een GD. Bijgevolg, wanneer een vrijgave wordt

aangegeven als GD; het gaat automatisch de beperkte onderhoudsfase in. Tijdens deze fase wordt de wijziging van de code door de techniek, inclusief insectenregelaars met belangrijke codeherwerking, strikt beperkt en gecontroleerd door een programmabeheerder. Dit waarborgt dat er geen slechte bug wordt toegevoegd aan een door GD gecertificeerd Cisco IOS-softwareversie. Een GD wordt bereikt door een bepaalde onderhoudsversie. De volgende onderhoudsupdates voor die vrijgave zijn ook GD-releases. Cisco IOS-softwarerelease 12.0 heeft bijvoorbeeld de GD-certificering op 12.0(8). Cisco IOS-softwarereleases 12.0(9), 12.0(10) enzovoort zijn IP-releases.

Experimentele of diagnostische afbeeldingen

Experimentele of diagnostische beelden worden soms technische specialiteiten genoemd en worden alleen gecreëerd wanneer belangrijke softwareproblemen zijn geïdentificeerd. Deze beelden maken geen deel uit van het normale vrijgaveproces. Afbeeldingen in deze categorie zijn klantspecifieke gebouwen die zijn ontworpen om te helpen bij het diagnosticeren van een probleem, om een bug-oplossing te testen of om een onmiddellijke oplossing te bieden. Er kan een onmiddellijke oplossing worden geboden wanneer het geen optie is om te wachten op de volgende tijdelijke of onderhoudsrelease. Experimentele of diagnostische beelden kunnen worden gebaseerd op elke ondersteunde softwarebasis, met inbegrip van onderhouds- of voorlopige versies van elk releasetype. Er bestaan geen officiële naamgevingsconventies, maar in veel gevallen zal de developer initialen, exp (bij wijze van experiment) of extra cijfers aan de achternaam van de basisafbeelding toevoegen. Deze afbeeldingen worden alleen op een tijdelijke basis ondersteund, in combinatie met Cisco-ontwikkeling, omdat de Cisco TAC- en Cisco IOS-release bewerkingen geen ondersteuning bieden voor documentatie zoals symbolische tabellen of basisafbeeldingsgeschiedenis. Deze beelden worden niet door Cisco intern getest.

Levenscycli release-mijlpalen

Op een bepaald moment worden GD-releases vervangen door nieuwe releases met de laatste netwerktechnologieën. Daarom is een vrijstellingstijdverkorting vastgesteld met de volgende drie belangrijke mijlpalen:

- **End of Sales (EOS)**—Voor grote releases is de EOS-datum drie jaar na de datum van eerste commerciële verzending (FCS). Dit stelt een einddatum vast waarvoor de release voor nieuwe systemen kan worden aangeschaft. De EOS-release blijft beschikbaar voor het downloaden van Cisco Connection Online (CCO) voor onderhoudsupgrades.
- **End-of-Engineering (EOE)** - De EOE-release is de laatste onderhoudsrelease voor de GD-release en volgt doorgaans ongeveer drie maanden na de EOS-release. Klanten kunnen technische ondersteuning blijven ontvangen via Cisco TAC en de EOE-release downloaden van CCO. Het productbericht, waarin de EOS- en EOE-releases en de datums worden aangekondigd, wordt één jaar voor de geplande EOS-datum gepubliceerd. Op dit moment zouden klanten moeten beginnen om het verbeteren van hun IOS van Cisco software te onderzoeken om voordeel te halen uit de nieuwste netwerktechnologieën.
- **End-of-life (End-of-life)** - aan het einde van de release en de levenscyclus wordt alle ondersteuning voor de Cisco IOS-softwarerelease beëindigd en niet langer beschikbaar voor downloads op de end-of-life datum. In het algemeen is de datum van exportgerichte bedrijven vijf jaar na de datum van exportgerichte bedrijven. Een EOL-productbulletin wordt ongeveer één jaar vóór de werkelijke EOL-datum gepubliceerd.

Cisco IOS-conventie voor nummering

De Cisco IOS-afbeeldingsconventie biedt een volledig profiel van alle vrijgegeven afbeeldingen. De naam omvat altijd de belangrijkste release-ID en de onderhoudsidentificatiecode. De naam kan ook een treinaanwijzer, een herbouwer (voor de onderhoudsvrijgave), een bedrijfsonderdeel (BU) - specifieke eigenschapaanwijzers en herbouwingsidenten van de BU omvatten. Het formaat kan als volgt worden verdeeld:

Sectie Namen van conventie	verklaring
x.y	Een combinatie van twee afzonderlijke (één of twee) cijfers identificatoren gescheiden door een "." dat de hoofdreleasewaarde identificeert. Deze waarde wordt bepaald door Cisco IOS-marketing. Voorbeeld: 12.1
z	Een tot drie cijfers die de onderhoudsrelease van x.y identificeren. Dit gebeurt elke acht weken. De waarden zijn 0 bij bèta, 1 bij FCS en 2 voor de eerste onderhoudsrelease. Voorbeeld: 12.1(2)
p	Eén alpha-teken dat een heropbouw van x.y(z) identificeert. De waarde begint met een kleine letter "a" voor de eerste heropbouw, dan "b", enzovoort. Voorbeeld: 12.1(2a)

A	Een tot drie letters is de aanwijzer van de vrijlatingstrein en zijn verplicht voor CTED, STED en X releases. Het identificeert ook een familie van producten of platforms. Technologie-versies maken gebruik van twee letters. De eerste letter vertegenwoordigt de technologie en de tweede letter wordt gebruikt voor differentiatie. Bijvoorbeeld:
	`A = Access Server/Dial technology (example:11.3AA)`
	`B = Broadband (example:12.2B)`
	`D = xDSL technology (example:12.2DA)`
	`E = Enterprise feature set (example:12.1E)`
	`H = SDH/SONET technology (example:11.3HA)`
	`N = Voice, Multimedia, Conference (example:11.3NA)`
	`M = Mobile (example:12.2MB)`
	`S = Service Provider (example:12.0S)`
	`T = Consolidated Technology (example:12.0T)`
	`W = ATM/LAN Switching/Layer 3 (example:12.0W5)`
	Een "X" in de eerste positie van de releasenaam identificeert een eenmalige vrijgave gebaseerd op de CTED "T"-trein. Bijvoorbeeld XA, XB, XC, etc. Een "X" of "Y" in de tweede positie van de releasenaam identificeert een kortstondige ED-release op basis van of verbonden met een STED-

	release. Bijvoorbeeld 11.3NX (gebaseerd op 11.3NA), 11.3WX (gebaseerd op 11.3WA), enzovoort.
o	Optioneel een of twee-cijferig getal dat een heropbouw van een bepaalde releasewaarde identificeert. Laat het leeg als het niet om een heropbouw gaat. Begint met 1, dan 2, enzovoort. Voorbeeld: 12.1(2)T1, 12.1(2)XE2
u	Een of twee-cijferig getal dat de functionaliteit van de BU-specifieke release identificeert. De waarde wordt bepaald door het marketingteam van de BU. Voorbeeld: 11.3(6)WA4, 12.0(1)W5
v	Een tot twee-cijferig getal aanwijzer die de onderhoudsvrijgave van de BU-specifieke code identificeert. De waarden zijn 0 bij bèta, 1 bij FCS en 2 als de eerste onderhoudsrelease. Voorbeeld: 11.3(6)WA4(9), 12.0(1)W5(6)
p	Eén alpha-karakter-aanwijzer die een heropbouw van een specifieke technologie-release identificeert. De waarde begint met een kleine letter "a" voor de eerste heropbouw, dan "b", enzovoort. Voorbeeld: 11.3(6)WA4(9a) zou een heropbouw van 11.3(6)WA4(9) zijn.

De volgende grafiek etiketteert de verschillende secties van de Cisco IOS naamgevingsconventie:

Bijlage B - Cisco IOS-betrouwbaarheid

De betrouwbaarheid van Cisco IOS is een gebied waar Cisco voortdurend probeert te verbeteren. Alvorens op klant georiënteerde best practices te bespreken, is enig begrip van de interne IOS van Cisco kwaliteits- en betrouwbaarheidsinspanningen nodig. Deze secties zijn vooral bedoeld om een overzicht te geven van de recentere inspanningen van Cisco in Cisco IOS softwarerelease en welke de aannames van klanten met betrekking tot softwarebetrouwbaarheid zouden moeten worden gemaakt.

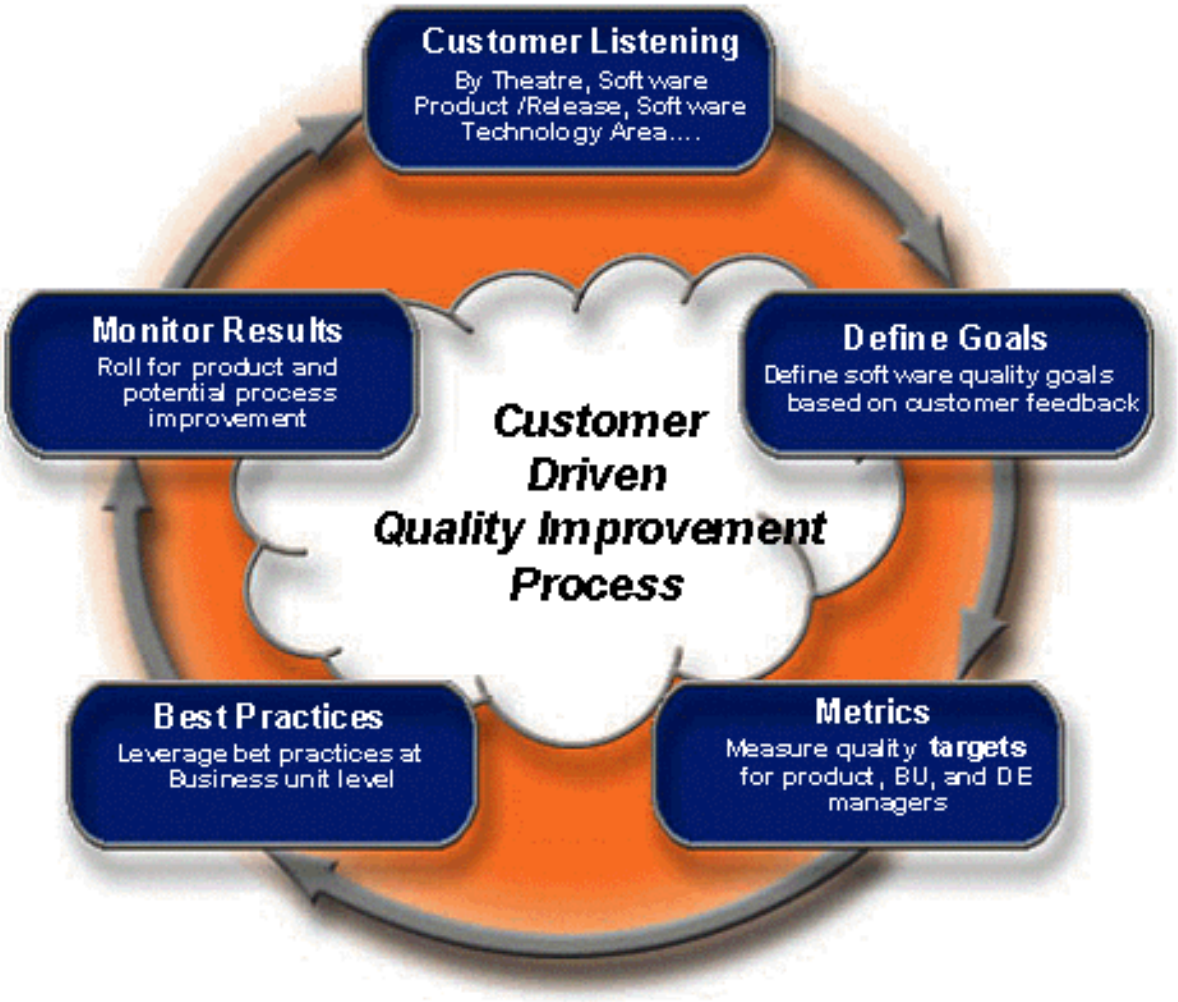
Cisco IOS Quality-of-Service

Cisco heeft een duidelijk gedefinieerd IOS-ontwikkelingsproces dat GEM Great Engineering Methodology (GEM) wordt genoemd. Dit proces heeft een levenscyclus van drie fasen:

- Strategie en planning
- Uitvoering
- Plaatsing

De algemene gebieden in de levenscyclus omvatten prioritering van de introductie van functies, ontwikkeling, het testproces, de introductie van software, de eerste klant Shipped (FCS), GD, en duurzame engineering. Cisco volgt ook een aantal best practice-richtlijnen van software van organisaties zoals International Standards Organisation (ISO), Telcordia (voorheen Bellcore), IEEE en het Carnegie Mellon Software Engineering Institute. Deze richtlijnen worden in de GEM-processen van Cisco opgenomen. Cisco-softwareontwikkelingsprocessen zijn gecertificeerd overeenkomstig ISO 9001 (1994).

Het primaire proces voor de verbetering van de software van Cisco IOS is een klantgericht proces waardoor Cisco naar klanten luistert, doelstellingen en metriek definieert, best practices implementeert en resultaten controleert. Dit proces wordt gedreven door een organisatorisch team dat zich inzet om de softwarekwaliteit te verbeteren. Een diagram van het Cisco IOS-kwaliteitsverbeteringsproces wordt hieronder weergegeven:



Het kwaliteitsverbeteringsproces heeft verschillende meetbare doelstellingen voor het begrotingsjaar 2002 en daarna. Deze doelstellingen zijn in de eerste plaats gericht op het terugdringen van tekortkomingen door softwareproblemen eerder in de testcyclus te identificeren, de achterstand te verminderen, de consistentie van de functies en de helderheid van de

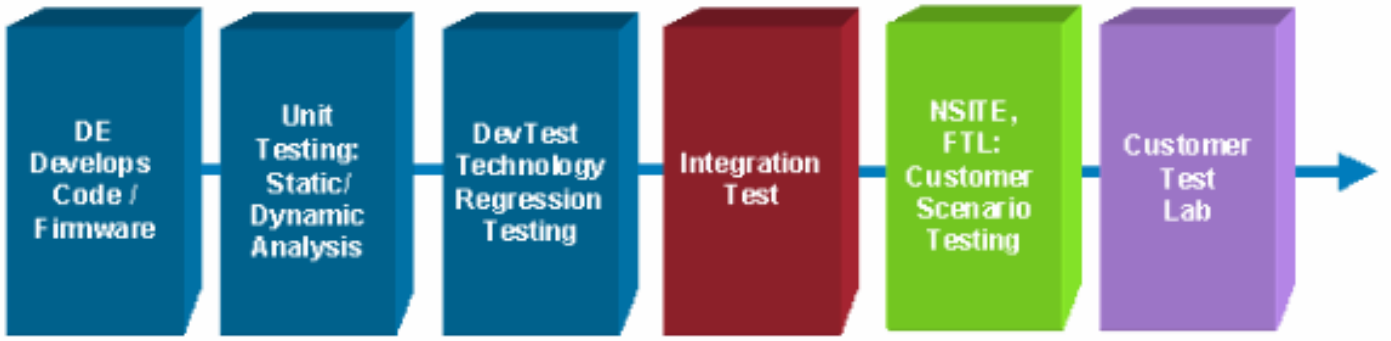
softwarerelease te verbeteren, en consistente voorspelbare vrijgaveprogramma's en softwarekwaliteit te leveren. Initiatieven om deze gebieden aan te pakken zijn nieuwe gereedschappen voor testdekking (het identificeren van gebieden met een zwakkere testdekking), verbetering van het testproces en verbeteringen van de Cisco IOS-systeemregressie. Er zijn extra bronnen toegepast om deze kwesties aan te pakken en er is een executive en cross-functionele toezegging voor alle primaire Cisco IOS-softwarereleases.

Cisco IOS-release testen

Een integraal deel van de de kwaliteitsinspanning van de softwarebetrouwbaarheid binnen Cisco is de kwaliteit, het bereik en de dekking van testen. Over het geheel genomen heeft Cisco de volgende IOS kwaliteitsdoelstellingen:

- Verminder gevonden Cisco interne regressie defecten. Dit houdt onder meer in dat de ontwikkeling van een hoger niveau moet worden verbeterd en dat meer problemen bij de statische/dynamische analyse moeten worden onderkend.
- Beperk klantfouten
- Totaal uitstaande tekortkomingen verminderen
- Vergroot de helderheid en consistentie van de softwarerelease
- Functie- en onderhoudsreleases met planning en kwaliteit leveren

Cisco interne testen kunnen worden gezien als een proces waarbij verschillende defecten worden geïdentificeerd in verschillende fasen van het testen. Het algemene doel is het vinden van de juiste soorten defecten in het juiste lab. Dit is om verschillende redenen belangrijk. Het eerste en belangrijkste is dat in latere testfasen wellicht geen adequate testdekking bestaat. De testkosten stijgen ook aanzienlijk van stadium tot stadium door het vermogen om in eerdere fasen te automatiseren en door de toenemende complexiteit en deskundigheid die later nodig zijn. In het volgende schema wordt het testspectrum voor Cisco IOS weergegeven.



De eerste fase is de ontwikkeling van software. Cisco heeft verschillende inspanningen op dit gebied om de initiële softwarefwaliteit te verbeteren. Ontwikkelingsgroepen voeren ook codebeoordelingen of zelfs meerdere codebeoordelingen uit om ervoor te zorgen dat andere ontwikkelaars softwareveranderingen of nieuwe functiecode goedkeuren.

De volgende fase is het testen van eenheden. Eenheidstests maken gebruik van instrumenten die softwareinteractie onderzoeken zonder gebruik van een lab. DevTest is een laboratoriumtest waarbij de functie/functionaliteit wordt getest en regressie wordt getest. Functie/functionaliteit testen is ontworpen om de functionaliteit van een bepaalde functie te onderzoeken. Dit omvat configuratie, desconfiguratie en het testen van alle functiestoornissen zoals gedefinieerd in de functiespecificatie. Regressietests worden uitgevoerd in een geautomatiseerde testfaciliteit die ontworpen is om functionaliteit en gedrag continu te valideren. Het testen is primair gericht op het routeren, het overschakelen, en de eigenschappen functionaliteit in een aantal verschillende

netwerktopologieën die pings en beperkte verkeersproductie gebruiken. Regressie testen wordt alleen uitgevoerd op een beperkte combinatie van functies, platforms, softwareversies en topologieën vanwege het extreem aantal mogelijke mutaties. Maar er worden vandaag meer dan 4000 testscripts voor regressie gebruikt. Het testen van de integratie is ontworpen om de mogelijkheden van laboratoriumtesten uit te breiden voor een uitgebreidere reeks producten en interoperabiliteit. Integratietests vergroten ook de dekking van de testcodes door het uitbreiden van tests tot interoperabiliteitstests, stress- en prestatietests, systeemtests en negatieve tests (testen van onverwachte gebeurtenissen).

De volgende labfase biedt end-to-end testen voor gebruikelijke klanten omgevingen. Deze worden in het bovenstaande schema weergegeven als Financial Test Lab (FTL) en NSITE, Customer Scenario Testing. FTL werd opgezet om tests te doen voor de missie cruciale financiële gemeenschap. NSITE is een groep die diepgaander testen voor verschillende Cisco IOS-technologieën biedt. De NSITE en FTL labs richten zich op gebieden zoals schaalbaarheid en prestatietests, upgradbaarheid, beschikbaarheid en veerkracht, interoperabiliteit en bruikbaarheid. De dienstbaarheid concentreert zich op kwesties van bulkvoorziening, gebeurtenis beheer/correlatie en het oplossen bij lading. Er bestaan andere laboratoria binnen Cisco voor verschillende verticale markten om deze gebieden te helpen testen.

Het laatste lab dat in het bovenstaande diagram wordt getoond, wordt geïdentificeerd als het klantlab. Klantentesten is een uitbreiding van de kwaliteitsinspanning en aanbevolen voor omgevingen met hoge beschikbaarheid om te verzekeren dat de exacte combinatie van functies, configuratie, platforms, modules en topologie volledig is getest. De testdekking moet netwerkschaalbaarheid en prestaties in de geïdentificeerde topologie, specifieke toepassingstests, negatieve tests in de geïdentificeerde configuratie, interoperabiliteitstests voor niet-Cisco-apparaten en ingebouwde tests omvatten.

Software MTBF

Een van de meest gebruikelijke maatstaven voor algehele betrouwbaarheid is de gemiddelde tijd tussen het falen (MTBF). MTBF's voor softwarebetrouwbaarheid zijn nuttig vanwege de analysecapaciteiten die zijn ontwikkeld voor de betrouwbaarheid van hardware met behulp van MTBF's. De betrouwbaarheid van de hardware kan nauwkeuriger worden bepaald aan de hand van bepaalde bestaande standaarden. Cisco gebruikt de methode van de onderdelentelling gebaseerd op de standaard MTBF gegevens van Telcordia Technologies. MTBF - software beschikt echter niet over overeenkomstige analysemethoden en moet voor MTBF - analyse een meetveld gebruiken.

De afgelopen drie jaar heeft Cisco veldmetingen van de softwarebetrouwbaarheid uitgevoerd voor het Cisco interne IT-netwerk en dit werk is gedocumenteerd in Cisco. Het werk is gebaseerd op softwaregedwongen crashes voor Cisco IOS-apparaten, die kunnen worden gemeten met behulp van SNMP-beknelinformatie en uptime informatie voor netwerkbeheer. De studie identificeert de betrouwbaarheid van de software met behulp van een statistisch lognormaal distributiemodel voor de geïdentificeerde softwarereleases. De gemiddelde tijd om software te repareren (MTTR) is gebaseerd op de gemiddelde tijd voor herstart en herstel van de router. Een hersteltijd van zes minuten wordt gebruikt voor bedrijfsomgevingen en vijftien minuten wordt gebruikt voor grotere internetserviceproviders (ISP's). Het resultaat van deze doorlopende studie is dat software over het algemeen voldoet aan een verfijnde beschikbaarheid na vrijgave of na een paar onderhoudsversies, en in de loop der tijd zelfs nog hoger is, gemeten met behulp van software-gedwongen crashes als de enige downtime-bron. De studie identificeerde potentiële MTBF-waarden als een bereik van 5000 uur voor vroege implementatiesoftware tot 50.000 uur voor algemene implementatiesoftware.

De meest algemene weerlegging voor dit werk is dat softwaregedwongen crashes niet alle tijden omvatten die zijn ontstaan als gevolg van problemen met softwarebetrouwbaarheid. Als deze maatstaf wordt gebruikt bij pogingen om de kwaliteit te verbeteren, kan hij het aantal softwaregedwongen crashes verbeteren maar kan hij andere belangrijke gebieden van softwarebetrouwbaarheid negeren. Deze opmerking blijft grotendeels onbeantwoord vanwege de moeilijkheid om met behulp van een statistische methodologie de betrouwbaarheid van de software nauwkeurig te voorspellen. De statistici van de softwarekwaliteit van Cisco zijn tot de conclusie gekomen dat een grotere steekproef van accurate gegevens nodig zou zijn om op betrouwbare wijze software MTBF te voorspellen met behulp van een breder scala aan outage types. Bovendien zou de theoretische statistische analyse moeilijk zijn vanwege variabelen zoals netwerkcomplexiteit, deskundigheid van personeel om softwaregerelateerde problemen op te lossen, netwerkontwerp, mogelijkheden en softwarebeheerprocessen.

Op dit moment is geen enkel industrieel werk verricht om de betrouwbaarheid van software nauwkeuriger te voorspellen met metingen in het veld, omdat het moeilijk is dit type gevoelige gegevens nauwkeurig te verzamelen. De meeste klanten willen ook niet dat Cisco de beschikbare informatie direct van hun netwerk verzamelt door de bedrijfseigen aard van beschikbaarheidsgegevens. Sommige organisaties verzamelen echter wel gegevens over de betrouwbaarheid van de software en Cisco moedigt organisaties aan om parameters over de beschikbaarheid te verzamelen vanwege softwareuitvallen en om een analyse van de oorzaak van de uitval uit te voeren. Organisaties met een hogere betrouwbaarheid van de software hebben deze proactieve houding gebruikt om de betrouwbaarheid van de software te verbeteren door middel van een aantal praktijken die zij kunnen beheersen.

Aannames voor softwarebetrouwbaarheid

Als resultaat van feedback van de klant, zijn er proactieve studies uitgevoerd door de Cisco IOS Technologies group en analyse van de basisoorzaak uitgevoerd door het Cisco Advanced Services team, een aantal nieuwere aannames en beste praktijken gevormd die helpen om de betrouwbaarheid van de software te verbeteren. Deze aannames betreffen het testen van verantwoordelijkheden, de looptijd of leeftijd van software, mogelijkheden en het aantal gebruikte softwareversies.

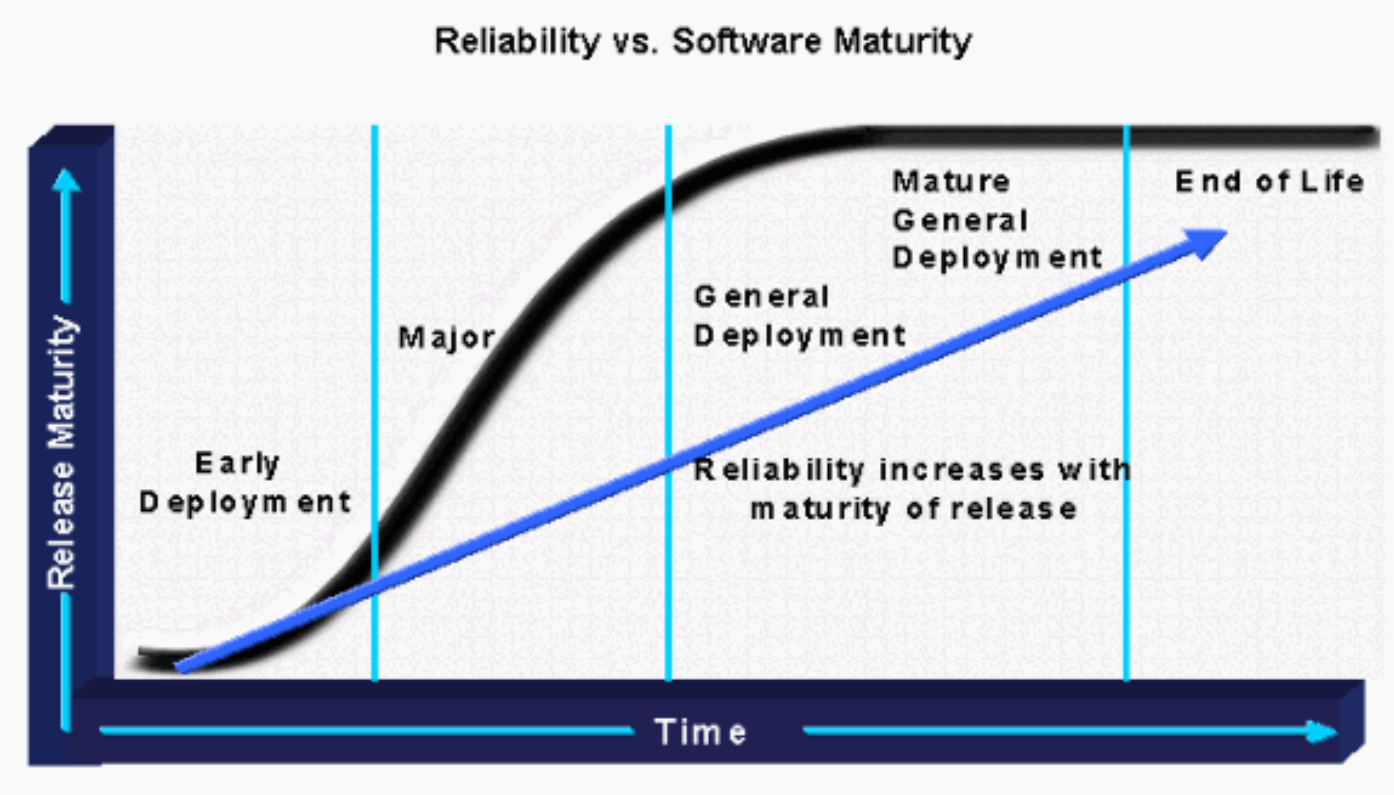
verantwoordelijkheid testen

De eerste nieuwe aanname betreft de verantwoordelijkheid voor het testen. Cisco is altijd verantwoordelijk voor het testen/valideren van nieuwe functies en functies om er zeker van te zijn dat ze in nieuwe producten werken. Cisco is ook verantwoordelijk voor regressie-testen om er zeker van te zijn dat de nieuwe softwareversies compatibel zijn met de software. Nochtans, kan Cisco niet elke eigenschap, topologie, en platform tegen elk potentieel voorbehoud valideren dat een klantomgeving kan brengen om te dragen (ontwerp idiosyncrasies, lading en verkeersprofielen). Hoge beschikbaarheid best practices voor klanten omvatten testen in een samengevouwen labtopologie die het productienetwerk bootst met behulp van door de klant gedefinieerde functies, ontwerp, services en toepassingsverkeer.

Betrouwbaarheid vs. softwarerelease

Softwarebetrouwbaarheid is voornamelijk een factor voor de looptijd van de software. Software rijpt naarmate hij blootstelling (gebruik) krijgt en zoals de geïdentificeerde insecten worden gecorrigeerd. De releaseactiviteiten van Cisco zijn naar een architectuur van treinrelease gegaan om ervoor te zorgen dat de software rijdt zonder dat er nieuwe functies worden toegevoegd. Klanten die een hoge beschikbaarheid nodig hebben, zoeken naar meer volwassen software met

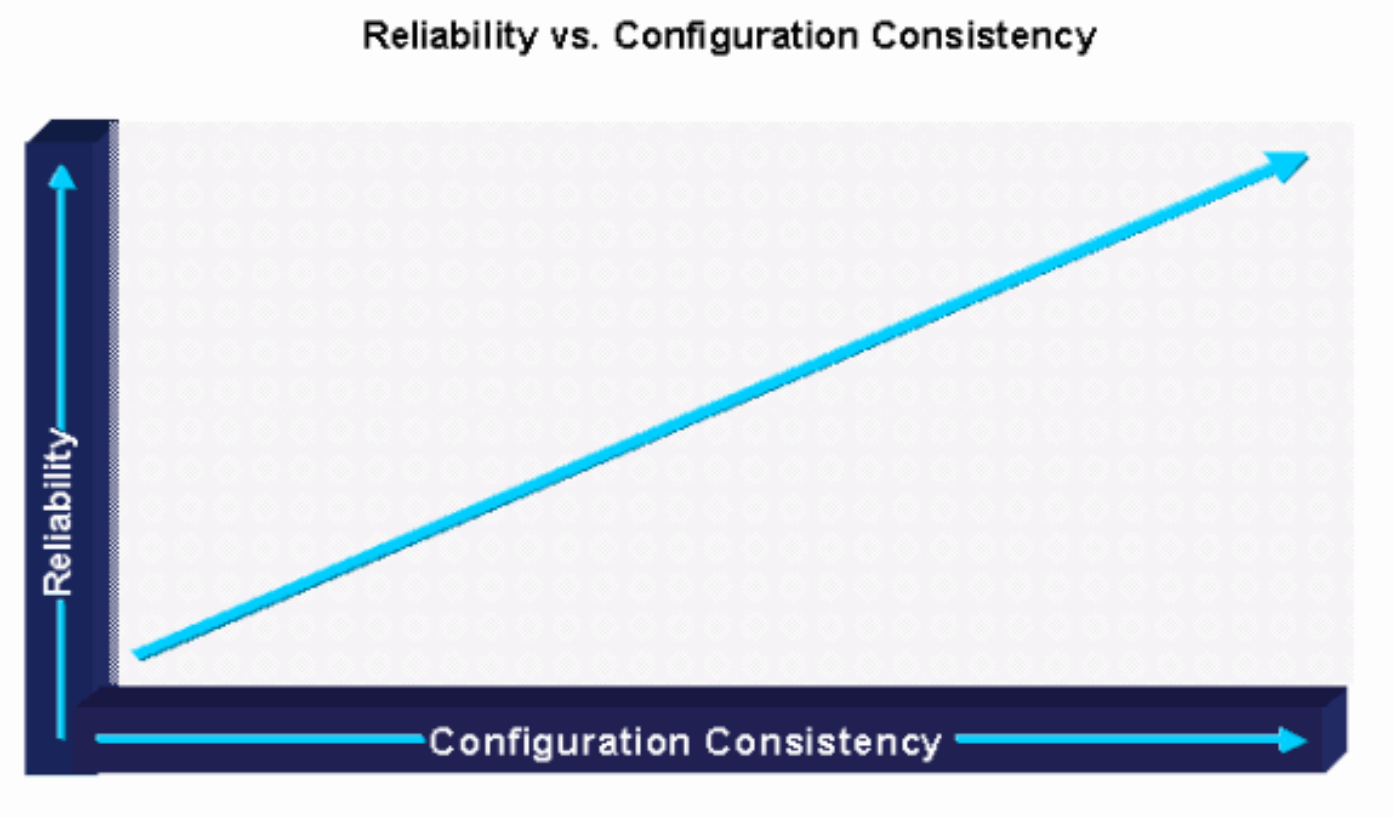
de functies die ze nu nodig hebben. Er bestaat dan een ruil tussen de looptijd van de software, de beschikbaarheid van vereisten en de zakelijke drijfveren voor nieuwe functies of functies. Veel organisaties hebben standaarden of richtlijnen voor aanvaardbare rijpheid. Sommigen zullen slechts de vijfde voorlopige vrijgave van een bepaalde trein accepteren. Voor anderen kan het de negende of de negende certificering zijn. Uiteindelijk moet de organisatie beslissen over het aanvaardbare risiconiveau in termen van softwarerelease.



Betrouwbaarheid vs. Hoeveelheid functies en standaarden

De betrouwbaarheid van de software is ook een factor van hoeveel van de code in een productieomgeving wordt getest en uitgeoefend. Aangezien de hoeveelheid verschillende hardwareplatforms en -modules toeneemt, neemt ook het gebruikte bedrag aan code toe, waardoor de blootstelling aan softwaredefecten over het algemeen toeneemt. Hetzelfde kan worden gezegd voor de hoeveelheid geconfigureerd protocollen, de verscheidenheid aan configuraties en zelfs de verscheidenheid aan toegepaste topologieën of ontwerpen. Design, configuratie, protocollen en hardwaremodulatiefactoren kunnen bijdragen aan de hoeveelheid code die wordt uitgeoefend en aan het verhoogde risico of de blootstelling aan softwaredefecten.

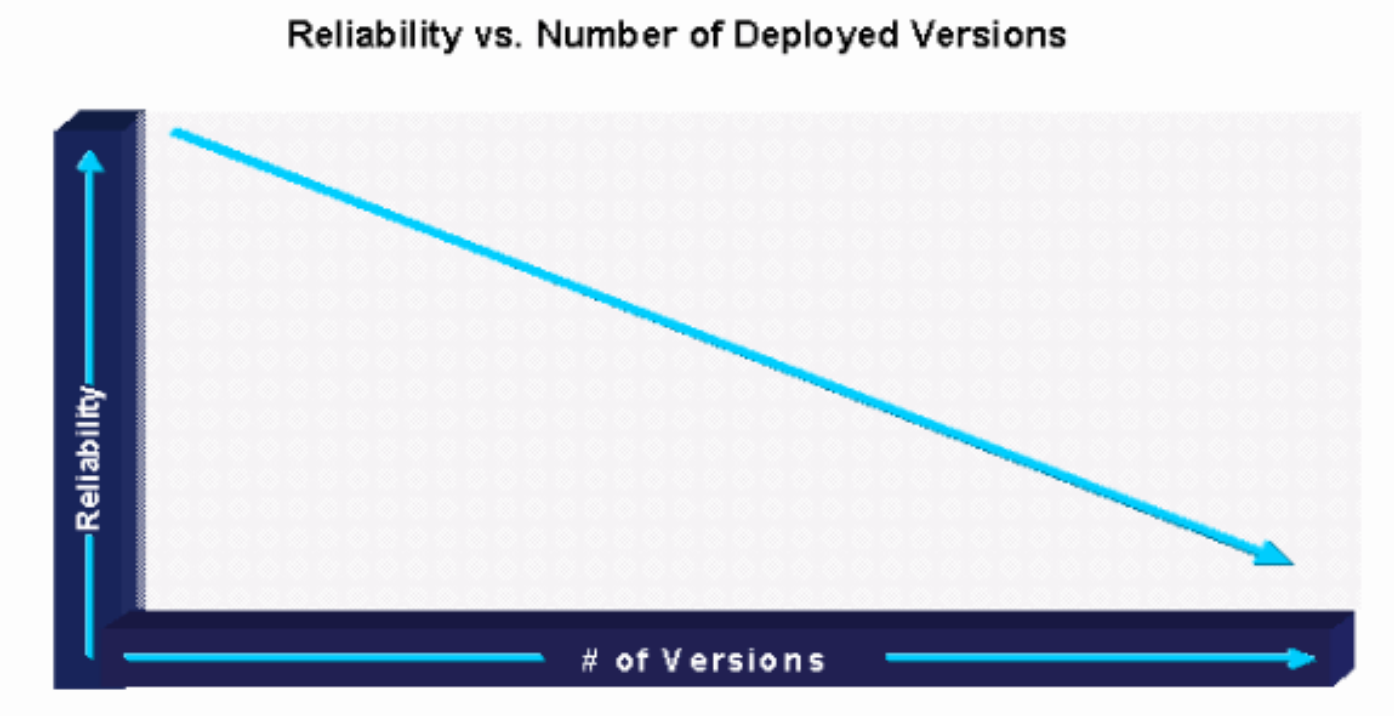
SOFTWARERELEVERWERKINGEN HEBBEN nu speciale software die de code over het algemeen beperkt die beschikbaar is in één specifiek gebied. Zakelijke eenheden hebben aanbevolen ontwerpen en configuraties die grondiger worden getest binnen Cisco en meer worden gebruikt door klanten. Klanten zijn ook begonnen met het toepassen van beste praktijken voor gestandaardiseerde modulaire topologieën en standaardconfiguraties om de hoeveelheid niet-geteste codeblootstelling te beperken en de algehele betrouwbaarheid van software te verbeteren. Sommige hoogbeschikbare netwerken hebben strikte standaardconfiguratierichtlijnen, modulaire topologienormen, en de controle van de softwareversie om het risico van niet-geteste codeblootstelling te verminderen.



Betrouwbaarheid vs. aantal uitgevoerde versies

Een andere factor voor de betrouwbaarheid van software is de interoperabiliteit tussen versies en de enorme hoeveelheid code die met meerdere versies wordt uitgeoefend. Aangezien het aantal softwareversies toeneemt, stijgt het gebruikte bedrag aan code ook, waardoor de blootstelling aan softwaredefecten toeneemt. Het betrouwbaarheidsrisico neemt bijna exponentieel toe door extra code die met meerdere versies wordt uitgeoefend. Inmiddels wordt erkend dat organisaties minstens een handvol versies in het netwerk moeten uitvoeren om specifieke functies en platformvereisten te dekken. Het uitvoeren van meer dan 50 versies in een grotendeels homogene netwerkomgeving is echter normaal gezien een aanwijzing voor softwareproblemen omdat deze vele versies niet goed kunnen worden geanalyseerd of gevalideerd.

Om de betrouwbaarheid van de software te verbeteren, voert de ontwikkeling van Cisco het testen van de softwareregressie uit om te verzekeren dat verschillende softwareversies compatibel zijn. Bovendien is de softwarecode modulair en is het minder waarschijnlijk dat kernmodules tussen versies in de loop der tijd aanzienlijk zullen veranderen. De release bewerkingen van Cisco hebben ook de hoeveelheid software veranderd die voor klanten beschikbaar is, omdat versies met bekende defecten of interoperabiliteitsproblemen snel van CCO worden verwijderd omdat defecten worden gevonden.



Gerelateerde informatie

- Cisco Internetworking-besturingssystemen (IOS)
- Technische ondersteuning en documentatie – Cisco Systems