IT Infrastructure Management

Week 6

1.Discuss in detail about the Release Management.

Ans;

Release management refers to the process of planning, designing, scheduling, testing, deploying, and controlling software releases.

It ensures that release teams efficiently deliver the applications and upgrades required by the business while maintaining the integrity of the existing production environment.

In the competitive, dynamic, and fluid world of business and IT, half-baked releases are the last thing you need.

The modern enterprise is a truly dynamic environment; and not all these changes are happening at the same pace. IT organizations need a way to orchestrate these myriad changes.

Release Management in ITIL

Release and Deployment management is one of the main processes under the Service Transition section of the Information Technology Infrastructure Library (ITIL) framework.

ITIL is the most widely adopted framework for the governance of technology products and services.

It helps organizations deliver their products and services in a quality-driven, customer-centric and cost-conscious way.

Release the Management Process

The specific steps of release management will vary depending on the unique dynamics of each organization or application. Nevertheless, the following sequence is the most common.

Request;

Release management starts with requests for new features or changes to existing functions. There's no guarantee that all requests made will eventually translate into a new release. Each request is evaluated for its rationale, feasibility, and whether there's a way to fulfill it by reconfiguring the application version already in production.

Plan;

This is the most important step in a release's evolution. It's here that the release's structure is defined. A robust plan ensures the release team stays on track and that

requirements are satisfied. Create or reuse a workflow or checklist that can be referred to by stakeholders throughout the release process. The workflow should detail not just scope and milestones but responsibilities.

Design and Build;

This is the programming phase where the requirements are converted to code. The release is designed and built into executable software.

Testing;

Once the release is deemed ready for testing, it's deployed to a test environment where it's subjected to non-functional and functional testing (including user acceptance testing or UAT).

If bugs are found, it's sent back to developers for tweaking then subjected to testing again. This iterative process continues until the release is cleared for production deployment by both the development team and the product owner.

Deployment;

The release is implemented in the live environment and made available to users. Deployment is more than just installing the release. It entails educating users on the changes and training them on how to operate the system in the context of the new features.

Post-Deployment;

Post-deployment, the release moves to the support phase where any bugs are recorded that will eventually necessitate a request for changes. The cycle thus begins again.

Release Management Success Indicators

For a release to be deemed successful, it must attain the following objectives:

- It's deployed on time.
- It's deployed within budget.
- It has little to no impact on current users.
- It satisfies the needs of current and new users, technological advances and/or competitive demands.

2.Explain: Mean Time To Repair-MTTR & Mean Time Between Failures-MTBF.

Ans;

MTTR:

Anytime you see the phrase "mean time to," it means you're looking at the average time between two events. Mean time to repair (MTTR) is a metric used by maintenance

departments to measure the average time needed to determine the cause of and fix failed equipment.

It gives a snapshot of how quickly the maintenance team can respond to and repair unplanned breakdowns.

It's important to remember the MTTR calculation considers the period of time between the beginning of the incident to the time the equipment or system returns to production. This includes:

- Notifying maintenance technicians
- Diagnosing the issue
- Fixing the issue
- Reassembling, aligning and validating equipment
- Resetting, testing and starting up the equipment or system for production

The MTTR formula does not take into account lead time for spare parts and is not meant to be used for planned maintenance tasks or shutdowns.

MTTR, as it pertains to maintenance, is a good baseline for figuring out how to increase efficiency and limit unplanned downtime, therefore saving money on the bottom line.

It also highlights why repairs might be taking longer than normal, which, when addressed, can get critical equipment up and running fast, minimizing missed orders and increasing customer service.

In the interest of efficiency, MTTR analysis provides insight into how your team purchases equipment, schedules maintenance and handles maintenance tasks.

Even though MTTR is considered reactive maintenance, tracking MTTR gives you a look into how effective and efficient your preventive maintenance program and tasks are.

For example, equipment with a lengthy repair time might have underlying root causes that contribute to the failure.

MTTR can help you start investigating the root cause of failures and get you on your way to a solution. For example, if you notice MTTR increasing in a particular asset, it may be due to the fact that preventive maintenance tasks aren't standardized.

A technician might get a work order telling him to lubricate a certain part, but it may not lay out which lubricant to use or how much, leading to further equipment failures.

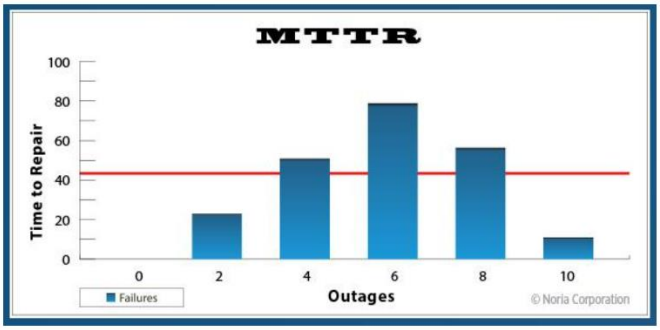
MTTR analysis is also helpful when it comes to making decisions on whether to repair or replace an asset. If a piece of equipment takes longer to repair as it gets older, it might be more economical to replace it.

MTTR history can also be used to help predict lifecycle costs of new equipment or systems.

MTTR Calculation:

A simple example of MTTR might look like this: if you have a pump that fails four times in one workday and you spend an hour repairing each of those instances of failure, your MTTR would be 15 minutes (60 minutes / 4 = 15 minutes).

Another example could involve an asset that experiences 10 outages in a 90-day period. The outage times (time of detection to time the asset is back to production) are 24, 51, 79, 56 and 12 minutes. The MTTR for this 90-day period is 44 minutes. That is the average time between the detection of the issue to the recovery of the asset.



MTBF:

MTBF is used to anticipate how likely an asset is to fail within a certain time period or how often a certain type of failure may occur.

When paired with other maintenance strategies, like failure codes and root cause analysis, and additional maintenance metrics, like MTTR, it will help you avoid costly breakdowns.

MTBF mean for maintenance:

Failure is a problem and knowing everything about a problem is often the best way to solve it.

Measuring MTBF is one way to get more information about a failure and mitigate its impact. Conducting an MTBF analysis helps your maintenance team reduce downtime while saving money and working faster.

3.What are the main problems that occur while implementing configuration management function?

Ans;

Networks form an integral part of our technology-driven era. Configurations in networks play a critical role as they keep networks fully functional and free from any downtime.

It is also challenging and hectic to manage them. As a result, a lot of errors and downtime occur because of misconfigurations or unauthorized configuration changes.

Almost 80% of network downtime is due to configuration-related errors.

A small change to a configuration could cause an entire IT infrastructure to fall in a minute, leading to huge losses in terms of money and time.

Here are five key configuration management challenges that a network admin faces in managing configuration changes and how ManageEngine Network Configuration Manager helps you overcome them.

1. Identifying major security flaws and their criticality

Security is the key to everything. Leave one gate open, and an entire empire falls. Blocking unauthorized outside traffic from the internal network and using only firewalls for safety is no longer adequate for protecting data as many threats will still make it into the network.

Security misconfigurations made it onto the 2021 OWASP Top 10 list of most critical web application security risks.

For example, imagine that in a huge organization with multiple teams and networks, an unauthorized change makes it into a network.

Imagine the loss of time and money it could cause. The organization may not recover. If an authorized network operator in this same organization makes an unwanted change, it would cause the same impact as the unauthorized change.

Three major security concerns arise: who, what, and when. Whoever makes the change, be it authorized or unauthorized, needs to be tracked along with what the change is and when it was made.

2. Making configuration changes at regular intervals and measuring their impact

Technologies are always changing along with networks. Network admins need to stay up-to-date with the latest improvements and features, which tends to be difficult and tedious.

If there is a stable configuration, network admins tend to make changes to it on a hunch, which may lead to a positive change or a negative change. Sometimes, even bulk changes are made.

A change made to a good configuration can be for various reasons, such as to improve security or to make the configuration feasible.

If it becomes a negative change, network admins should be able to tackle it, even when it is severe and of unknown technological use.

3. Comparing configurations without automation

A lot of changes take place when configurations are constantly updated, and in most cases, admins want to refer to previous configurations to see what kinds of changes are needed.

In some cases, when a change turns out to be unwanted, comparing all the configuration changes made in that version with an old, stable version will help admins analyze the situation better and pinpoint unwanted changes during troubleshooting.

If there are numerous configuration changes made in a network, network admins cannot manually compare them all line by line.

It would be nearly impossible, tedious, and a huge loss of time. Some organizations still do manual configuration comparison, which takes a toll on their admins since it involves a lot of time and effort. This needs to be rectified immediately with automated configuration comparison.

4. Validating every change made to configurations

Network operators reporting to network admins make necessary changes to configurations for agile performance.

These changes, though authorized, need to be validated by network admins. Validating every change is a time-consuming task, and if something is not validated, it could lead to an unwanted change and downtime.

Thus, this is a major challenge of configuration change management to overcome.

5. Keeping tabs on real-time changes without change notifications

Whenever an unauthorized change is made and gets into a network, it is because the network admins were not aware of it.

Without real-time change notifications, admins will not know what changes are made or when.

When an unauthorized change is made, it needs to be tracked and should not be let in.

With a change notification feature, you get real-time notifications when a change is made, and the admin does not even need to be logged in to a network automation tool.

When this feature is not provided, it is a tedious challenge for admins to keep track of changes 24/7.

challenges of configuration management :

- Network Configuration Manager has multiple security layers to keep track of unauthorized entries and block them. It also has alarms if anything suspicious occurs.
- Network Configuration Manager provides flexibility by automating tasks. Admins will not have to learn difficult technologies or programming.
- On top of this, it separates positive changes and negative changes to save your network from downtime proactively.
- Network Configuration Manager has a feature called Diff View that provides side-by-side comparisons of two configurations on a single device or two configurations on two different devices.
- It is also color-coded for swift, agile comparisons by admins so they can easily see what has been changed and by whom.
- All changes need to be validated, even those from a reliable, authorized source. Network Configuration Manager has configuration change management, consisting of role-based access control, giving admins full authority to validate and even reject changes.

- No change made by operators goes past admins undetected. Thorough validation is done before changes are implemented.
- Everything in Network Configuration Manager is user-friendly, with no visibility challenges.
- It tracks all changes and keeps admins updated with real-time change notifications. It supports notifications via email, SNMP traps, syslog, and tickets.
- Unless an admin ignores a notification, there is no chance of an unauthorized change occurring.

4.Explain the process involved in Availability Management for an IT organization.

Ans;

Availability Management in ITIL is the ability of a configuration item or IT Service to perform its agreed function when required.

Availability is determined by reliability, maintainability, serviceability, performance, and security. It is the duty of availability management to make sure that the level of availability that is delivered in all the IT services fulfills the availability needs in a manner that is both timely and cost-effective.

Its main concern is to meet the present and future availability needs of the business

Process Activities of Availability Management

The key activities performed by availability management are:

- To determine the availability requirements from the business
- Monitor, measure, analyze, report, and review the availability of services and components.
- To perform an unavailability analysis that investigates all the events, incidents, and problems that involve unavailability. Necessary corrective actions are taken after this.
- Perform a service failure analysis where the underlying causes for service interruption are identified.
- To identify the vital business functions and design for availability and recovery.
- Perform component failure impact analysis (CFIA), single point of failure analysis (SPOF), and fault tree analysis (FTA).
- To create models to determine whether the new models will meet the stated requirements.