

The background of the image consists of several overlapping geometric shapes in various shades of blue, ranging from a light sky blue to a dark navy blue. The shapes are primarily triangles and quadrilaterals, creating a dynamic, abstract composition. The text 'APNIC' is positioned in the lower right area, set against a dark blue background.

APNIC

Network Monitoring and Management Module : Ticketing System

Objectives



Audience will get to know -

- a) What is a ticketing system and why it is important?
- b) The standard workflow of a ticketing system
- c) Generic features and Components of a TS
- d) How a TS can save time and complexity, of network operations.
- f) Recommendation and best practices of production deployment
- g) Deployment and integration
- h) Analytics and reporting

What is Ticketing System?



A **ticketing system** is a solution that automates processes, categorizes events, and store relevant data to ensure better reports and improved customer services.

It can be used for internal IT Support, Infrastructure support escalation, or for external customer support, or for all of them at a time.

In general, the first known ticketing system was used in ancient Greece for theater audiences.

What were the motivations to impose a ticket system at that time?

- Only an authorized person can get into the theater
- Authority can have the audience count
- And it may, if they need any other facilities, like fruits/drinks, can avail it by showing the tickets

Life without ticketing system



Without a ticketing system, what are the matters an organization may have to deal with?

- Telephone conversation or email correspondence
- Issues are recorded in emails or on spreadsheets.
- Managers don't know who is responsible for an event.
- Customers are not updated about their issues.
- Staff go on holiday or quit without handing over cases.
- Problems remain unsolved due to poor management.

Why ticketing system is necessary



Let us discuss a story of a service provider company –

- One internet user complained about a slower browsing experience through an email to the helpdesk.
- An engineer was assigned, based on the duty schedule, containing the client complaint and user information.
- Engineer has communicated with the internet user, but got another person of that house or organization other than the complainer, checked his given information, asked for a few more technical details.
- And after a few moments, the engineer rebooted the gateway devices, which has resolved the problem.
- Engineer has closed the ticket, mentioning his steps to resolve the issue.

Why ticketing system is necessary



After a few days, the same user complains about the same experience through an email.

- As usual a ticket was assigned to the available engineer.
- The responsible engineer checked the previous support history, and then talked with the person who made the complaint and get to know the experiences in detail.
- Without following those previous steps, the currently assigned engineer checked the network flow and captured packets.
- And by analyzing this information, he found out some strange activities were running, which was not supposed to be.
- Based on his/her analysis, he/she recommended some best practices and request to maintain internet usage hygiene.

Why ticketing system is necessary



Based on the above story, what statistics can we get?

- The user complaint history, how many issues they have logged.
- How much time was spent in the support process.
- Which engineer has the efficiency to solve an issue.
- What are the options, an engineer has to use to conduct productive support.
- Does the assigned engineers maintain the SOP (Standard Operation Policy)
- The support provided to the customer was compliant with SLA (Service Level Agreement)

Other sources of Information for ticket creation



Systems can report faults too!

- Smokeping increased ICMP latency or loss alerts
- SNMP interface or disk nearing capacity alerts
- Service not running alerts
- OSPF flapping alerts
- BGP lost peer alerts

Ticketing System Flowchart

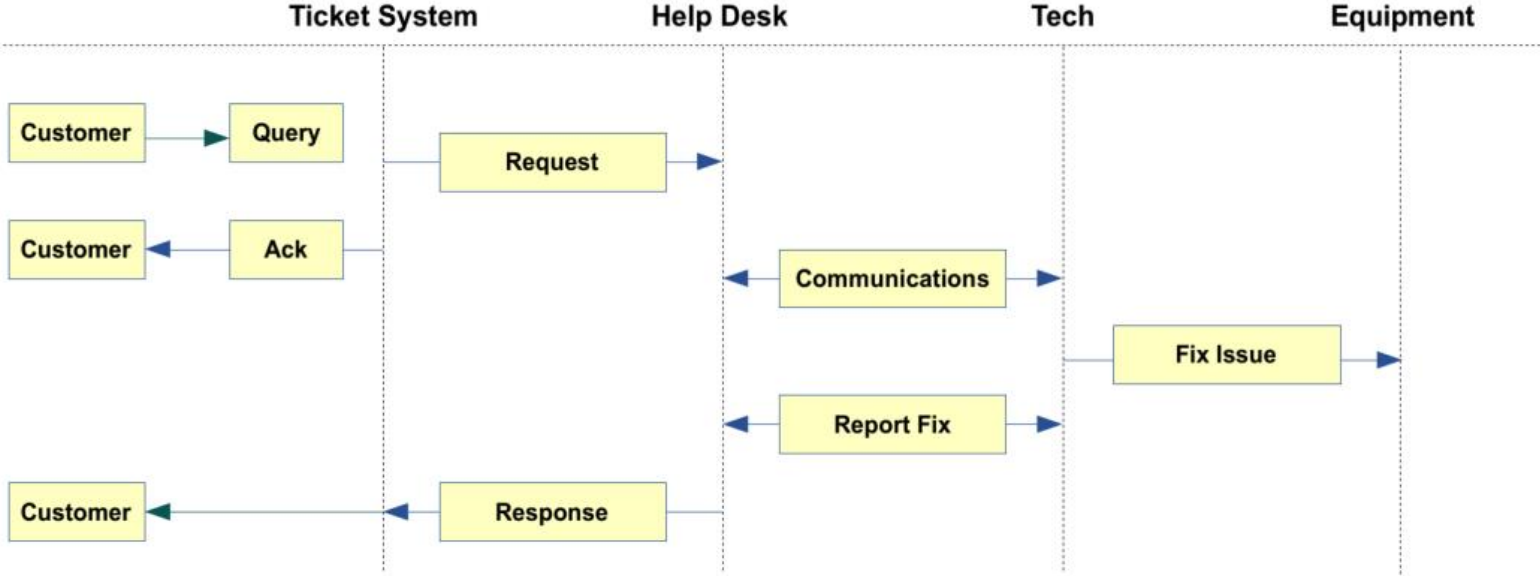


Image created by Jonathan Brewer for nsrc.org, licensed CC-BY-NC

Component of a Good Ticketing System



An efficient ticketing system needs the following –

- APIs (Application Programming Interfaces) & Email integration
- Users, Groups, and Roles
- Search
- Workflows
- Reporting and Analytics

Component of a Good Ticketing System



Users, Groups, and Roles

- Users are individual people, or sometimes applications
- Groups are collections of people or applications
 - Groups are important because no ticket should depend on a single person
 - Example groups could be help desk, network operations, sales
- Roles are sets of permissions that can be granted to Users or Groups
 - Sales might have permission to read, but not update, helpdesk tickets
 - Help desk might be able to to read, not update, network ops tickets

Component of a Good Ticketing System



Workflows

The sequence of industrial, administrative, or other processes through which a piece of work passes from initiation to completion.

- Ticket systems can pass a ticket through a workflow
- To aid a business process
- To ensure service level agreements are met
- To auto-escalate tickets that are not being attended

Component of a Good Ticketing System



APIs & Email Integration

APIs = Application Programming Interfaces

- Inbound integration = applications submitting tickets
 - Smokeping, Nagios/Icinga, LibreNMS
 - Email!
- Outbound integration = Ticket System calling other apps
 - Slack Channel Alerts
 - SMS or API based alerting like Pagerduty
 - Email!

Component of a Good Ticketing System



Reporting & Analytics

Help desk systems can reveal a lot about how a business is operating.

- Are some customers filing more tickets than others?
- Are some services less reliable than others?
- How long does it take to resolve a ticket:
 - For a particular class of user
 - For a particular product
- Are some teams better at resolving tickets quickly?

Well known Ticketing System



- osTicket, cloud based or on-prem – <https://osticket.com/>
- Request Tracker RT, on-prem – <https://bestpractical.squarespace.com/>
- OTRS, on-prem – <https://otrs.com/>
- Spiceworks, cloud or on-prem – <https://www.spiceworks.com/>

Features of osTicket



The first release of osTicket was in 2003.

- Tickets are created through email, and by helpdesk staff via form or direct call from a customer.
- Auto-response can be defined with a custom template.
- With flexible settings email alerts and notifications can be sent to update the correspondence.
- Can be integrated with other monitoring and alert tools
- OpenSource software, with regular maintenance
- Paid support and services are available
- Build on LAMP technologies



- To host a service, few things are needed to calculate.
 - The data volume to store
 - The session to manage
- Access control can be defined
 - If the system is only for internal use.
 - Or it is going to be used for external customer as well.
- Security Best practices should be defined as it is been deployed on LAMP.
- Admin user should have two-way authentication mechanism imposed.
- Monitoring of process running, services metrics and resource utilization is mandatory.

Thank You!

