06 SERVICE OPERATION PHASE

Service Operation adalah tahapan siklus hidup layanan TI yang mencakup semua kegiatan operasional harian pengelolaan layanan-layanan TI. Inilah saatnya semua produk strategi, desain, dan transisi layanan benar-benar dirasakan oleh pelanggan. Service Operation bertanggung jawab mengoperasikan layanan dan menjaga layanan TI yang dimiliki tetap hidup (live) dan bekerja sesuai dengan harapan dan kebutuhan pengguna/ pelanggan. Service Operation juga merupakan tahapan dimana pengguna dapat merasakan dan mengukur manfaat (value) yang diperoleh dari layanan TI.

Service Operation mencakup semua aktivitas yang diperlukan untuk mempertahankan layanan TI agar dapat terus bekerja di sepanjang waktu operasional layanan dan memberikan dukungan kepada pelanggan apabila mereka membutuhkan. Untuk itu, staf operasional layanan TI harus memiliki proses-proses dan peralatan pendukung untuk memonitor unjuk kerja layanan TI dan mendeteksi ancaman atau potensi kegagalan layanan TI.

Tujuan dan Cakupan Service Operation

Terdapat dua tujuan Service Operation, yaitu:

1. **Pengoperasian layanan TI**: mengkoordinasikan dan melaksanakan kegiatan dan proses yang dibutuhkan untuk memberikan layanan TI kepada pengguna dan pelanggan, serta mengelola layanan untuk memenuhi tingkat layanan yang telah disepakati.
2. **Pengelolaan teknologi pendukung layanan TI** (on-going management): mengelola teknologi yang digunakan untuk menghasilkan dan mendukung layanan TI. Oleh karena itu, sebagian besar aktivitas dari Service Operation adalah bagaimana memahami dan mengelola komponen-komponen teknologi, seperti server, mainframe, jaringan komputer, komunikasi, basis data, media penyimpanan, sistem desktop, dan aplikasi software. Aktivitas-aktivitas tersebut mencakup kegiatan pemantauan dan pengendalian untuk memastikan semua komponen bekerja sesuai target dan dapat menerima peringatan dini apabila terjadi kesalahan.

Dengan adanya Service Operation memberikan panduan bagaimana mengelola layanan TI secara efisien dan efektif, serta menjamin tingkat kinerja yang telah disepakati bersama pelanggan. Panduan-panduan berfungsi untuk menjaga kestabilan operasional layanan TI dan pengelolaan perubahan Service Design.

Cakupan Service Operation meliputi proses, fungsi (functions), organisasi, dan peralatan teknologi yang digunakan untuk menjalankan aktivitas-aktivitas yang dibutuhkan untuk penyediaan dan pendukung layanan, termasuk:

1. Layanan-layanan TI
2. Proses-proses manajemen layanan TI
3. Teknologi
4. Manusia

Prinsip-Prinsip Dasar Service Operation

Terdapat beberapa prinsip dasar yang harus dipenuhi pada saat pelaksanaan aktivitas Service Operation, yaitu:

1. **Keseimbangan yang tepat**
   Pada kenyataannya, setiap operasional layanan TI selalu akan menghadapi dilema antara fokus menjaga kestabilan infrastruktur layanan TI atau cepat menanggapi kebutuhan bisnis. Untuk menyediakan layanan TI yang stabil, penyedia layanan membutuhkan infrastruktur TI yang stabil dan staf yang berfokus untuk menjaganya. Di sisi lain, penyedia layanan juga harus responsif terhadap kebutuhan-kebutuhan bisnis yang umumnya membawa konsekuensi perubahan layanan TI dan infrastruktur TI tanpa mengganggu layanan bisnis. Keseimbangan yang harus dijaga ini meliputi:

a. **Internal IT View** versus **External Business View**
   Memastikan organisasi memiliki keseimbangan antara fokus teknologi dan pemahaman bahwa penyedia layanan TI adalah bagian dari layanan yang mendukung aktivitas bisnis.
b. **Stability** versus **Responsiveness**
   Menjaga keseimbangan antara menjaga infrastruktur yang stabil, namun di saat yang sama mampu merespon terhadap perubahan pada kebutuhan bisnis.
c. **Quality of Service** versus **Cost of Service**
   Seimbang dalam memastikan setiap keputusan layanan, dengan mempertimbangkan kepentingan kualitas, termasuk biaya.
d. **Reactive** versus **Proactive**
   Penyedia layanan harus seimbang dalam hal reaktif menindaklanjuti permintaan atau masalah layanan TI dan proaktif merencanakan perbaikan layanan TI.

2. **Komunikasi**
   Salah satu kunci utama keberhasilan penyediaan layanan operasional layanan TI yang baik adalah komunikasi. Komunikasi menjadi bagian yang penting dalam Service Operation karena menjadi media untuk menyampaikan dan mendukung layanan itu sendiri, mencakup di antaranya:
   a. Komunikasi operasional rutin
   b. Komunikasi antara shift kerja
   c. Laporan unjuk kerja
   d. Komunikasi dalam proyek-proyek TI
   e. Perubahan, pengecualian, dan tindakan darurat
   f. Training untuk proses-proses baru atau termodifikasi

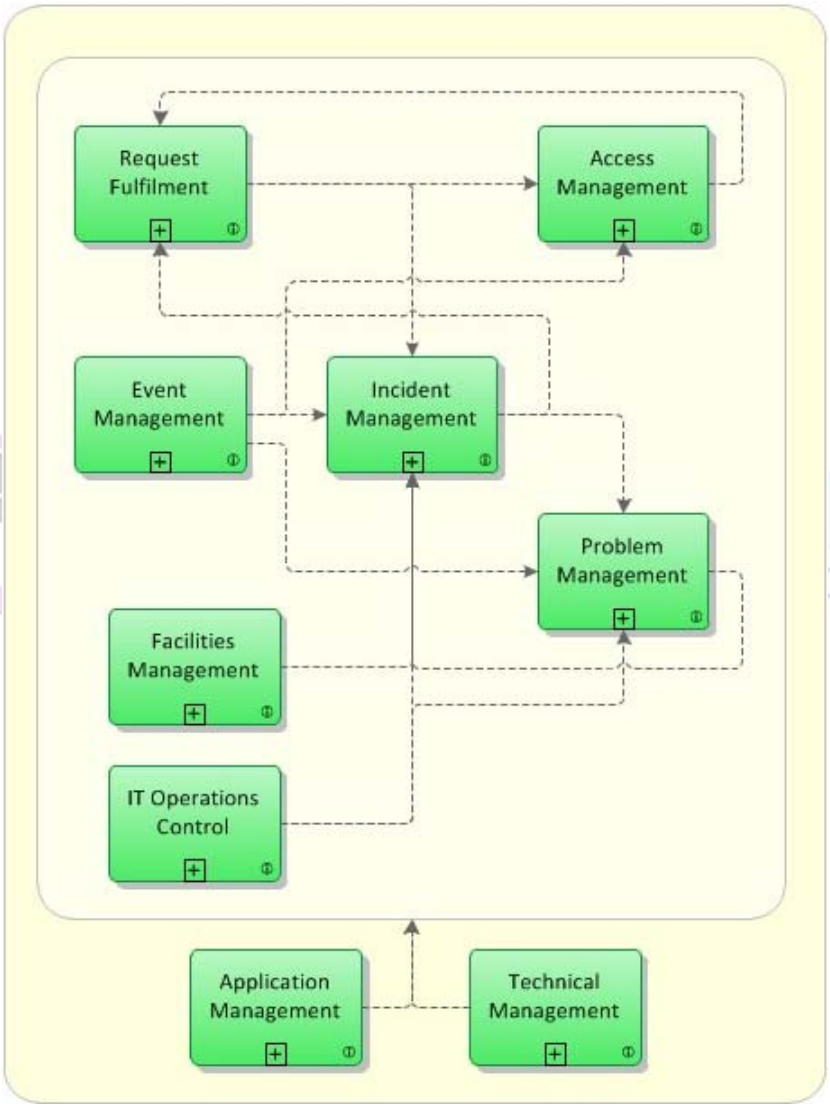
Interaksi dan komunikasi dalam Service Operation ini terjadi antar:
1. Anggota dalam tim TI yang sama atau antar tim TI yang berbeda
2. Departemen lain selain Departemen TI
3. Pengguna (users) dan pelanggan (customers)
4. Tim dan departemen Service Operation

Proses-Proses dalam Service Operation

Terdapat sembilan proses dalam Service Operation, yaitu:

1. **Event Management**
   Proses memastikan semua Configuration Item (CI) dan layanan TI yang sedang berjalan selalu termonitor, memfilter dan mengkategorisasi setiap kondisi/status (events) layanan TI untuk diambil tindakan yang tepat.
2. **Incident Management**
   Proses mengelola setiap insiden yang terjadi pada layanan TI agar layanan TI bagi pelanggan dapat segera pulih sesegera mungkin.
3. **Problem Management**
   Proses mengelola akar-akar masalah penyebab insiden layanan TI agar insiden-insiden tersebut tidak terjadi lagi di kemudian hari dan meminimalkan dampak dari insiden yang tidak dapat dicegah.
4. **Request Fulfilment**
   Proses memenuhi permintaan pelanggan layanan TI.
5. **Access Management**
   Proses memberikan hak akses layanan TI kepada pengguna yang berhak dan mencegah akses pengguna yang tidak berhak. Proses ini pada dasarnya adalah mengimplementasikan kebijakan-kebijakan yang telah dirumuskan di proses Information Security Management.
6. **IT Operations Control**
   Proses (sekaligus sebuah function/unit yang bertanggung jawab) memonitor dan mengontrol layanan-layanan TI dan infrastruktur pendukungnya. Proses ini berisi aktivitas-aktivitas operasional rutin harian terkait komponen-komponen infrastruktur dan aplikasi-aplikasi layanan TI, seperti pekerjaan penjadwalan, backup dan restore, manajemen print dan laporan, serta perawatan rutin.

7. **Application Management**
   Proses (sekaligus sebuah function/unit yang bertanggung jawab) mengelola aplikasi-aplikasi (software), seperti update, instalasi, dan pengembangan.
8. **Technical Management**
   Proses (sekaligus sebuah function/unit yang bertanggung jawab) menyediakan ahli dan dukungan teknis untuk pengelolaan infrastruktur TI.
9. **Facilities Management**
   Proses (sekaligus sebuah function/unit yang bertanggung jawab) mengelola dan merawat lingkungan fisik di mana infrastruktur TI berada, mencakup listrik, pendingin, manajemen akses bangunan, kebersihan, keamanan, dan monitoring lingkungan kerja.



Gambar 6.1 Hubungan Proses-Proses Service Operation

Event Management

Event Management adalah rangkaian aktivitas mendengarkan atau mendeteksi apapun pesan ketidaknormalan dari infrastruktur TI dan melakukan sesuatu untuk mencegah hal yang buruk terjadi dan berdampak kepada pengguna.

Event Management analoginya seperti proses mendengarkan pesan-pesan dari infrastruktur TI. Pada dasarnya semua komponen teknologi yang kita gunakan selalu berkomunikasi menyampaikan pesan kepada kita tentang kondisi mereka: apakah mereka masih bekerja, bekerja dengan baik, atau ada sesuatu yang salah dengan pola kerja dan kondisi mereka.

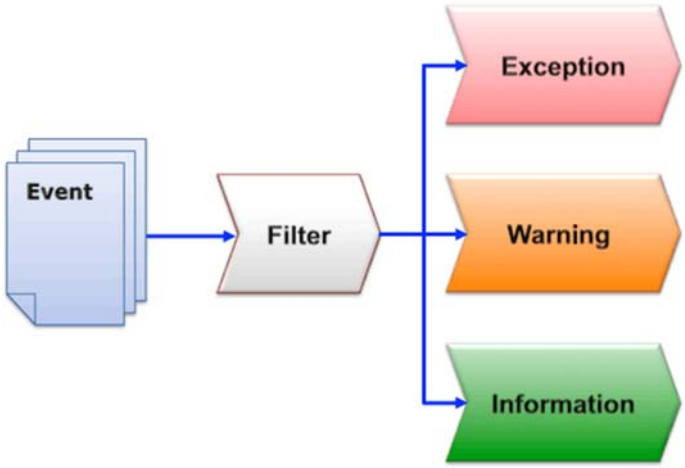
Event didefinisikan ITIL sebagai "Any change of state that has significance for the management of a Configuration Item (CI) or IT service" atau dapat diartikan sebagai sebuah perubahan keadaan pada infrastruktur TI yang memiliki nilai penting bagi manajemen layanan TI atau Configuration Item TI. Events biasanya pesan atau tampilan yang dihasilkan oleh layanan, Configuration Item, atau alat monitoring.

Event-event ini umumnya dideteksi dan dikenali melalui notifikasi-notifikasi yang ditampilkan oleh sebuah layanan TI, CI, atau tool monitoring. Aplikasi untuk mengotomatisasikan aktivitas-aktivitas dalam Event Management disebut Event Management tools, yang umumnya mampu memberikan informasi "Apa yang terjadi?", "Apa artinya?", dan "Apa yang harus dilakukan?".

Jenis-Jenis Event

Setiap event umumnya akan dideteksi dan dapat dikategorikan menjadi tiga, yaitu:

1. **Informasi (Information)**
   Sebuah event yang menunjukkan sesuatu yang diharapkan dan normal terjadi sehingga tidak memerlukan tindakan apapun, hanya dicatat sebagai file log.
   Contoh
   Kejadian proses backup data rutin selesai dilakukan, "printing 90%", atau seseorang login ke sistem. Kejadian tersebut dicatat dalam file log dan disimpan untuk jangka waktu yang telah ditentukan.
   Beberapa event digunakan untuk memeriksa status perangkat atau layanan, untuk mengkonfirmasi keadaan suatu kegiatan, atau untuk menghasilkan statistik.
   Contoh
   User login, pekerjaan batch selesai, perangkat power up, dan jumlah pengguna login ke dalam aplikasi.
2. **Peringatan (Warning)**
   Sebuah event yang memiliki nilai mencapai ambang batas. Sebuah respon tindakan dapat diperlukan/tidak diperlukan. Informasi peringatan ini dibutuhkan untuk mengambil tindakan yang diperlukan untuk mencegah potensi kegagalan.
   Contoh
   Informasi bahwa volume disk hanya tersisa 5% dari maksimum kapasitas yang tersedia.
3. **Ketidakwajaran (Exception)**
   Sebuah event yang menginformasikan bahwa sebuah layanan atau komponen berjalan tidak sewajarnya (abnormal). Sebuah tindakan biasanya dibutuhkan untuk merespon sebuah event exception.
   Contoh
   Jumlah pengguna yang melebihi batas atau kecepatan akses internet yang sangat lambat.



Gambar 6.2 Jenis-Jenis Event

Tujuan Event Management adalah untuk mengelola event, mencakup mendeteksi event, menganalisis event, dan menentukan tindakan yang paling sesuai untuk event tersebut. Menganalisis event terkait dengan mendefinisikan apakah event tersebut penting/tidak bagi manajemen layanan TI. Tindakan yang mungkin diambil di antaranya adalah mencatatnya/log, mengabaikannya, menyampaikan peringatan kepada orang lain, atau menindaklanjuti dengan tindakan Incident Management, Problem Management, atau Change Management.

Cakupan

Event Management dapat diimplementasikan untuk berbagai aspek manajemen layanan yang membutuhkan kontrol dan dapat diotomatisasi, termasuk di antaranya:

1. Configuration Items
2. Kondisi-kondisi lingkungan (seperti deteksi api atau asap)
3. Monitoring lisensi software yang terinstalasi di semua sistem
4. Keamanan sistem (seperti deteksi akses ilegal)
5. Aktivitas normal (seperti tracking penggunaan sebuah aplikasi tertentu atau unjuk kerja sebuah server)

Alert adalah istilah lain dalam Event Management yang berarti **informasi pemberitahuan** bahwa ambang batas telah tercapai, sesuatu telah berubah, atau sebuah kegagalan telah terjadi (warning atau exception). Alert dapat dalam bentuk pengiriman pesan ke monitor, telepon, atau pager, dan dihasilkan oleh peralatan/teknologi manajemen sistem serta dikelola oleh proses Event Management. Alert umumnya membutuhkan tindakan intervensi manusia, misalnya seorang teknisi untuk menginvestigasi sebuah event.

Tools

Peralatan teknologi informasi (software dan hardware) yang membantu dalam melakukan pekerjaan. Tool merupakan aspek penting untuk manajemen event yang efektif.

Contoh

Alat-alat monitoring, alat manajemen jaringan, dan alat-alat Event Management.

Gambar 6.3 Contoh Aplikasi/Tool Event Management

Aktivitas-Aktivitas pada Event Management

Aktivitas-aktivitas proses Event Management meliputi:

1. **Event notification, event detection, event logging**
   Pada dasarnya setiap komponen dalam infrastruktur layanan TI senantiasa menciptakan dan menyampaikan pesan, hanya saja kita yang mengabaikan atau tidak mengerti.
   a. **Event notification**
      Aktivitas menentukan "pesan" apa yang ingin dihasilkan dari komponen-komponen TI dan informasi-informasi apa saja yang terkandung di dalam pesan tersebut.
   b. **Event detection**
      Menentukan jenis notifikasi mana yang ingin dideteksi dan dikenali oleh tools dan menentukan apakah event akan dicatat di log atau tidak, serta bagaimana mencatatnya.
   c. **Event logging**
      Mencatat data di dalam tools yang mendeteksi event tersebut atau menghasilkan rekaman event dalam event logging tool tertentu.
2. **Event filtered, event correlation**
   Filtering adalah aktivitas mengklasifikasikan sebuah event sebagai sebuah informational, warning, atau exception, serta memutuskan apakah akan mengambil tindakan atau sekedar mencatatnya (log) dan mengabaikannya.
   Event correlation mengambil kesimpulan event sama yang terjadi berulang-ulang sehingga dapat dipahami dampak akumulasinya.

   Contoh

   Sebuah warning event mengingatkan bahwa bandwidth jaringan kita sudah mencapai ambang batas dan bagi kita event ini tidak terlalu penting untuk ditindaklanjuti. Namun, jika warning event ini terjadi berulang-ulang dalam satu hari (misalnya terjadi 200 kali), maka bisa jadi kita mengklasifikasikannya sebagai **exception event** atau **alert** yang membutuhkan tindak lanjut.
3. **Response selection**
   Menentukan jenis respon untuk setiap kemungkinan event, mencakup pilihan:
   a. **Auto Response:** mengatur sebuah peralatan melakukan respon otomatis tertentu khusus untuk sebuah event.

      Contoh

      Sebuah server diatur melakukan restart sendiri secara otomatis.
   b. **Alert:** sistem akan secara otomatis menghasilkan dan mengirimkan informasi peringatan untuk menarik perhatian.
   c. **Incident, Problem, Change:** berdasarkan kriteria-kriteria yang telah ditetapkan oleh manajemen, sistem Event Management diatur mampu secara otomatis menghasilkan laporan/permintaan untuk proses Incident Management, Problem Management, atau Change Management. Laporan atau permintaan proses ini masih perlu persetujuan manajemen untuk dilaksanakan.
4. **Review action, close event**
   Aktivitas me-review penanganan sebuah event perlu dilakukan untuk memastikan bahwa langkah yang tepat telah diambil. Mengingat jumlah event yang dihasilkan setiap hari begitu banyak dan tidak memungkinkan untuk me-review semua event yang tercatat, maka review dapat dilakukan secara acak atau hanya pada event-event yang penting. Aktivitas review ini juga penting dalam rangka melakukan analisis trend event-event yang terjadi.
   Setiap kali sebuah tindakan telah dilakukan terhadap sebuah event atau sekumpulan event, maka aktivitas event dapat dikatakan selesai/ditutup.

   Contoh

   Event-event informasi (informational) otomatis selesai jika telah tercatat (ter-log).

Incident Management

Incident adalah kejadian interupsi sebuah layanan TI yang tidak terencana (tidak diharapkan) atau penurunan kualitas sebuah layanan TI. Incident mencakup:

1. **Sebuah interupsi layanan TI yang tidak direncanakan sebelumnya**

Contoh

Koneksi internet tiba-tiba terputus atau jika SLA mengatakan layanan internet tersedia dari jam 8 pagi hingga 6 sore, namun ternyata jam 5 sore koneksi internet sudah tidak dapat diakses.

2. **Penurunan kualitas dari sebuah layanan TI**

Contoh

Kecepatan internet melambat dari biasanya.

3. **Kegagalan sebuah komponen infrastruktur TI (Configuration Item), meskipun belum berdampak pada layanan TI**

Contoh

Sebuah harddisk di server mirror rusak.

Incident Management adalah rangkaian aktivitas untuk mengatasi permasalahan layanan TI mengembalikan layanan TI agar berfungsi/bekerja kembali sesuai tingkat layanan yang telah disepakati. **Tujuan** proses Incident Management adalah:

1. Mengembalikan operasional normal layanan TI secepat mungkin
2. Meminimalkan dampak buruk gangguan layanan TI terhadap operasional bisnis
3. Memastikan standar kualitas layanan yang telah ditetapkan/disepakati dapat selalu terjaga

Incident Management **berbeda dengan Problem Management**. Incident Management fokus pada bagaimana mengembalikan layanan TI ke keadaan normal secepat mungkin, tanpa perlu mengetahui/mengatasi akar masalah kejadian gangguan tersebut. Jadi, Incident Management bertanggung jawab mengatasi gejala sebuah masalah, bukan penyebab dasar masalah. Problem Management adalah proses lain dalam Service Operation yang bertanggung jawab mengidentifikasi dan menyelesaikan akar permasalahan layanan TI.

Service Desk adalah pihak yang bertanggung jawab untuk penanganan setiap incident, dari ditemukan/ dilaporkan hingga dinyatakan selesai, meskipun melalui aktivitas eskalasi. Service Desk bertanggung jawab mengawal setiap progress penanganan, menginformasikannya kepada pengguna, hingga menutup laporan.

Contoh

Beberapa waktu yang lalu, koneksi internet di kantor mati karena kabel fiber optic yang putus digigit tikus. Untuk mengembalikan layanan internet secepat mungkin, staf help desk kantor telah menghubungi Telkom untuk memperbaiki sambungan kabel fiber optic tersebut. Tindakan perbaikan sambungan kabel fiber optic itu disebut sebagai Incident Management. Untuk mencegah kejadian kabel digigit tikus kembali, maka staf help desk memasang pelindung jalur kabel fiber optic tersebut dan memasang racun tikus. Tindakan ini masuk ke dalam kategori Problem Management.

Cakupan

Incident Management meliputi kejadian/situasi/event apapun yang mengganggu, atau berpotensi mengganggu sebuah layanan TI, baik yang disampaikan langsung oleh user ke service desk maupun yang dideteksi aplikasi-aplikasi tool Event Management dan Incident Management.

Beberapa istilah dalam Incident Management:

1. **Normal Service Operation**
   Sebuah kondisi operasional dimana layanan-layanan dan CI-CI bekerja sesuai dengan level layanan dan operasional yang telah disetujui.
2. **Problem**
   Akar penyebab satu atau lebih dari satu incident. Catatan sebuah problem (problem record) akan di-input-kan ke dalam tool aplikasi Problem Management pada saat penyedia layanan memutuskan bahwa akar penyebab satu/lebih incident perlu untuk diinvestigasi. Akar penyebab sebuah masalah incident seringkali belum diketahui pada saat sebuah problem record dibuat, proses Problem Management yang bertanggung jawab melakukan investigasi lanjut.
3. **Impact**
   Seberapa besar potensi kerugian yang ditimbulkan atau seberapa banyak jumlah pengguna yang terkena dampak dari sebuah accident, problem, atau change pada proses-proses bisnis. Umumnya impact diukur berdasarkan seberapa pemenuhan target/standar tingkat layanan TI (service levels) akan terpengaruh. Impact dan urgency digunakan untuk menentukan prioritas (urgency).

4. **Urgency**
   Seberapa lama waktu yang dibutuhkan dari sebuah kejadian accident, problem, atau change memiliki dampak signifikan bagi bisnis, atau seberapa cepat bisnis membutuhkan penyelesaian sebuah incident, atau seberapa lama penanganan sebuah accident dapat menunggu. Sebuah accident mungkin memiliki impact yang tinggi, namun memiliki urgency yang rendah.

   Contoh

   Sebuah aplikasi Sistem Informasi Perwalian Mahasiswa yang down/error memiliki impact tinggi bagi proses perwalian mahasiswa, namun urgency rendah karena belum akan berdampak hingga awal semester baru mulai kembali.

5. **Priority**
   Sebuah kategori yang digunakan untuk menentukan nilai penting sebuah incident, problem, atau change. Priority diukur berdasarkan imapct dan urgency, serta digunakan untuk menentukan seberapa cepat sebuah tindakan respon harus segera diambil.

   Contoh

   Dalam sebuah dokumen SLA dinyatakan bahwa sebuah incident "Prioritas 2" harus diatasi dalam waktu 12 jam sejak laporan masuk.

6. **Timescale**
   Target waktu respon dan penyelesaian sebuah incident sesuai dengan yang ditulis di dokumen SLA. Semua timescale harus telah disepakati untuk semua incident dalam SLA dan secara tertulis dinyatakan sebagai target-target dalam dokumen OLA dan UC. Semua kelompok/unit yang terkait dengan OLA dan UC harus mengetahui semua timescale ini. Aplikasi tool manajemen layanan dapat diimplementasikan untuk mengotomatisasikan timescale dan melakukan eskalasi incident apabila dibutuhkan sesuai dengan aturan penanganan incident yang telah ditetapkan.

7. **Major Incident**
   Kategori tertinggi sebuah incident, yang umumnya memiliki karakteristik sebagai berikut:
   a. Incident yang memiliki dampak besar pada bisnis
   b. Memiliki urgency tinggi
   c. Penyebab telah diketahui, tetapi belum ada panduan solusi atau workaround

   Perusahaan harus mendefinisikan indikator major incident (misalnya jika kejadian berdampak pada minimal 5000 pengguna) dan membuat prosedur penanganan khusus, baik dalam hal timescale (umumnya timescale lebih pendek) maupun staf yang menanganinya. Beberapa incident yang memiliki prioritas rendah namun berpotensi memiliki dampak besar bagi bisnis seringkali juga ditangani dengan prosedur major incidents.

8. **Incident Model**
   Sebuah prosedur standar (aktivitas dan timescale yang telah ditetapkan sebelumnya) yang dibuat untuk menangani incident "Standar" atau "Spesial".
   Incident standar adalah incident yang sering terjadi.

   Contoh

   Masalah terkait printer, incident model berupa SOP yang menjelaskan tingkat prioritas penanganan incident sekaligus nama staf yang menanganinya.

   Incident spesial misalnya incident terkait keamanan sistem, sehingga prosedur standar penanganannya harus diteruskan ke proses Information Security Management. Informasi yang harus ada dalam incident model mencakup:
   a. Tindakan-tindakan yang harus diambil untuk menangani incident
   b. Urutan tindakan
   c. Penanggung jawab
   d. Timescales
   e. Prosedur eskalasi

9. **Service Desk**
   Fungsi dalam perusahaan (di bawah Departemen TI) yang bertanggung jawab menangani proses Incident Management, termasuk aktivitas menerima laporan, pencatatan (log), mengkategorikan, memprioritaskan, dan menutup insiden.

10. **Incident Status Tracking**
    Semua incident harus dapat ditelusuri (di-track) statusnya agar dapat ditangani dengan tepat dan dilaporkan. Terdapat 4 (empat) kemungkinan status incident:

a. "Open", artinya incident sudah dikenali, tetapi belum ditangani
b. "In Progress", artinya incident dalam proses investigasi dan penyelesaian
c. "Resolved", artinya incident telah selesai ditangani, tetapi status operasional layanan normal belum dilaporkan/dikonfirmasi oleh pengguna (user)
d. "Closed", artinya pengguna (user) telah mengkonfirmasi bahwa incident telah diselesaikan dan kondisi operasional normal telah kembali

11. Workaround

Tindakan standar (terdokumentasi) untuk mengurangi dampak buruk dari sebuah incident atau problem yang belum diketahui solusi totalnya. Workaround untuk problem-problem TI didokumentasikan dalam known error records. Workaround untuk accident-accident yang tidak ada hubungannya dengan problem records didokumentasikan dalam incident record.

Contoh

Me-restart server yang sering tiba-tiba mati.

12. Known Error

Sebuah masalah (problem) yang telah diketahui dan terdokumentasi akar masalahnya, gejala-gejala incident-incident yang terkait, solusi standarnya atau langkah yang bisa dilakukan adalah meminimalkan dampak buruk apabila solusi totalnya belum diketahui (workaround-nya).

13. Known Error Database (KEDB)

Basis data known error yang akan dipergunakan oleh service desk dan staf pendukung lainnya untuk melakukan Incident Management dan workaround. Basis data ini dibuat oleh proses Problem Management dan digunakan oleh proses Incident Management dan Problem Management. KEDB seringkali menjadi bagian dari Configuration Management System atau disimpan di dalam Service Knowledge Management System.

14. Problem Model

Standard Operating Procedure (SOP) penanganan sebuah permasalahan tertentu.

15. Eskalasi (Escalation)

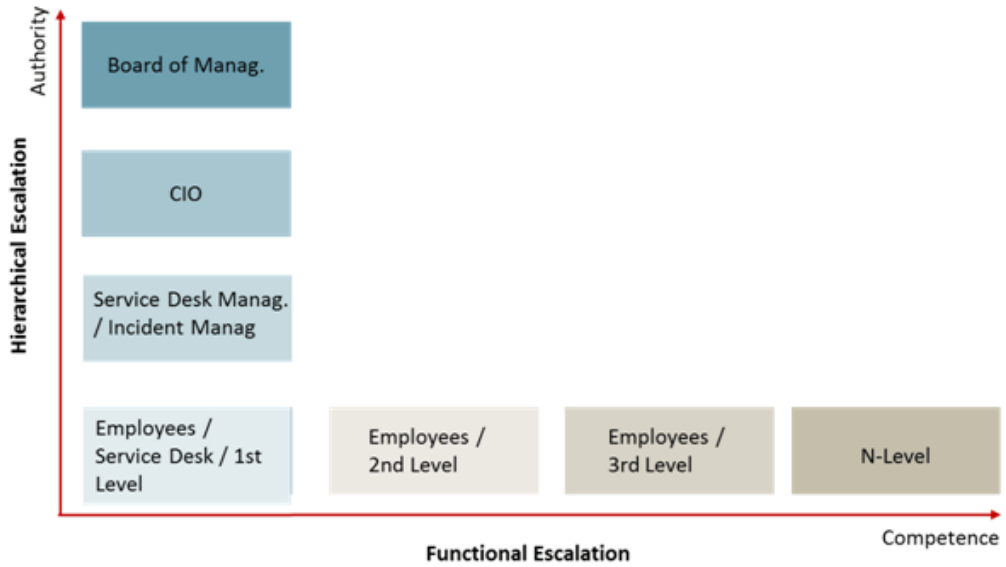
Tindakan meneruskan laporan dan penanganan incident yang belum terselesaikan ke fungsi lain dalam perusahaan untuk memperoleh dukungan lebih lanjut. Terdapat dua macam eskalasi:

a. Functional escalation

Meneruskan penanganan sebuah incident ke tim pendukung atau second level support (fungsi Technical Management atau Application Management).

b. Hierarchical escalation

Meneruskan/mengkomunikasikan incident yang belum terselesaikan ke manajemen bisnis di atasnya (misalnya manajer) untuk dicari solusinya.



Gambar 6.4 Jenis Eskalasi dalam Incident Management

Aktivitas-Aktivitas pada Incident Management

Terdapat enam aktivitas utama dalam Incident Management, yaitu:

Gambar 6.5 Aktivitas dalam Incident Management

1. **Identifikasi incident dan logging**

   Aktivitas **menemukan/mengenali sebuah incident**. Sebuah incident dapat diidentifikasi dari banyak hal, seperti:

   a. Dilaporkan oleh user kepada service desk, baik datang secara langsung atau via telepon
   b. Dilaporkan melalui e-mail
   c. Dilaporkan melalui formulir online di web
   d. Terdeteksi oleh proses Event Management

   Terlepas dari mana sebuah incident diidentifikasi, setiap incident harus **dicatat (di-log)** oleh service desk. Catatan incident ini harus mencatat detil incident (termasuk tanggal dan jam incident) serta senantiasa di-update selama aktivitas investigasi, penyelesaian, hingga saat penutupannya. Proses pencatatan ini juga mencakup aktivitas pemilihan kategori incident dan prioritas penanganannya. Untuk mempermudah melakukan semua aktivitas tersebut, biasanya service desk memiliki software aplikasi Incident Management yang disebut **Integrated IT Service Management Toolsets**.

2. **Kategorisasi incident**

   Incident yang dicatat selanjutnya dikategorisasi dengan **kode kategori incident**. Umumnya kategorisasi ini dimulai dari kategorisasi jenis layanan, komponen (CI), hingga spesifik incident-nya. Proses pencatatan ini dilakukan dengan memanfaatkan tool software.

   Aktivitas kategorisasi incident ini penting karena dapat membantu:

   a. Apabila incident tidak dapat diatasi oleh staf help desk, kategorisasi ini akan membantu keputusan kepada siapa incident ini akan diteruskan
   b. Sebagai salah satu dasar pertimbangan memutuskan prioritas penanganan incident
   c. Menyediakan masukan untuk analisis tren incident

3. **Prioritisasi incidents**

   Aktivitas selanjutnya adalah menentukan prioritas penanganan sebuah incident yang telah dicatat dan dikategorisasi, secara teknis dengan memberikan **kode prioritas** incident. Penentuan prioritas berdasarkan impact dan urgency incident tersebut. Prioritas ini menentukan urutan kapan incident ini harus segera ditangani dan diselesaikan. Setiap perusahaan seharusnya memiliki standar yang jelas tentang impact dan urgency sehingga prioritas penanganan incident dapat diambil lebih tepat dan terstandar.

Dalam menentukan skala prioritas incident, umumnya kode prioritas berupa angka 1, 2, 3, dan seterusnya, serta setiap prioritas diberikan deskripsi tingkat incident (critical, high, medium, low, planning), dan ditentukan skala waktu standar penanganannya.

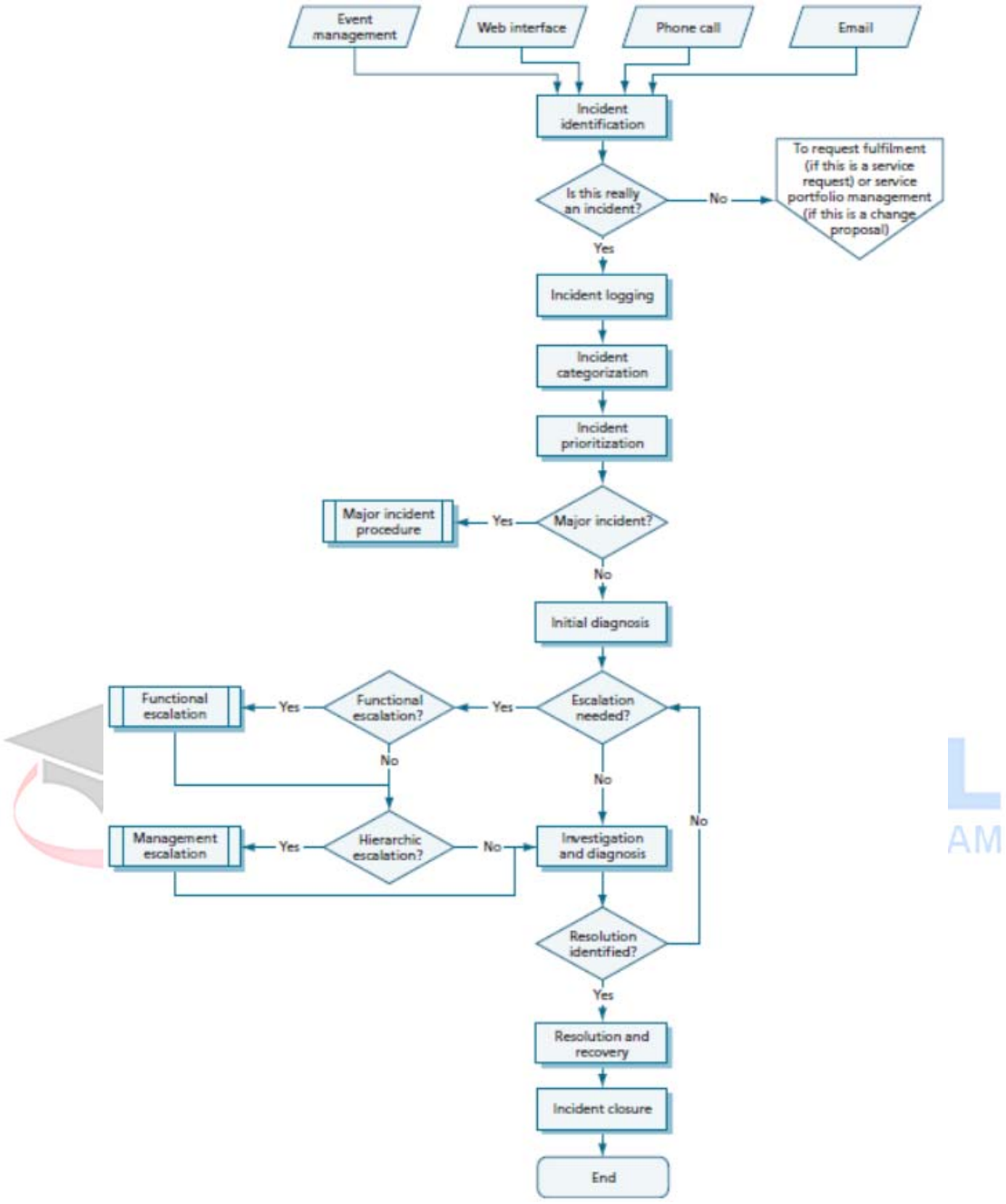
Contoh

Priority Code	Description	Target Resolution Time
1	Critical	1 hour
2	High	8 hour
3	Medium	24 hour
4	Low	48 hour
5	Planning	Planned

Seorang mahasiswa mendatangi ruangan service desk, melaporkan bahwa ia tidak dapat login ke dalam Sistem Informasi Akademik untuk melakukan registrasi ulang. Saat staf service desk mencatat laporan incident ini ke dalam software Incident Management, secara default tertulis prioritas 3 (Medium). Namun, saat mahasiswa tersebut melaporkan bahwa masalah yang sama juga dihadapi oleh 35 temannya yang lain dan staf service desk mencatatnya ke dalam software, maka prioritas meningkat menjadi 2 (High). Selanjutnya, mahasiswa juga menginformasikan bahwa batas waktu registrasi ulang hanya tinggal hari tersebut sehingga incident harus ditangani dan diselesaikan hari itu juga, maka staf help desk menaikkan prioritas incident menjadi 1 (Critical), yang berarti harus diselesaikan dalam waktu 1 (satu) jam sesuai SLA.

4. **Diagnosis awal**
   Service desk akan berupaya untuk menyelesaikan incident terlebih dahulu sebelum diteruskan kepada tim teknis. Pada aktivitas ini, staf service desk juga mengumpulkan semua informasi dari pengguna terkait incident, termasuk kegiatan pengguna yang berujung pada incident tersebut dan ciri-ciri incident. Selanjutnya, staf service desk mencari solusinya, umumnya dengan mengakses informasi di known error database (KEDB).
5. **Investigasi dan diagnosis; penyelesaian masalah dan pemulihan kondisi layanan (Resolution and Recovery)**
   Apabila staf service desk mengetahui solusi dari incident, maka selanjutnya ia langsung dapat menyelesaikannya sendiri. Namun, jika service desk tidak mampu menyelesaikan sebuah incident atau target waktu penyelesaian incident oleh service desk telah terlewati, maka incident harus segera **dieskalasi** untuk memperoleh dukungan lebih lanjut. Umumnya, tahap pertama eskalasi adalah **Functional Escalation**, baru apabila sebuah incident tidak dapat diselesaikan baik oleh service desk maupun tim pendukung, maka dilakukan **Hierarchical Escalation**.
   Dalam aktivitas ini, **investigation and diagnosis** berarti mencari tahu apa yang salah dan mencari solusi incident, sedangkan **resolution and recovery** berarti solusi incident telah diketahui serta selanjutnya diaplikasikan dan diuji coba. Tindakan pengembalian layanan harus dipastikan benar-benar tuntas dan catatan incident harus di-update.
   Sekali lagi, target utama proses Incident Management adalah mengembalikan layanan seperti semula atau sesuai SLA kembali, bukan menyelesaikan akar permasalahan layanan TI.
6. **Penutupan incident**
   Setelah incident diatasi, staf service desk harus menginformasikan kepada pengguna bahwa masalah telah teratasi dan memastikan pengguna puas dengan penanganan (misalnya dengan survei) serta setuju laporan incident ditutup. Umumnya cara terbaik menghubungi pengguna adalah dengan telepon.

Gambar berikut ini menunjukkan flowchart standar penanganan Incident Management. Service desk adalah pihak yang bertanggung jawab untuk penanganan setiap incident, dari ditemukan/dilaporkan hingga dinyatakan selesai, meskipun melalui aktivitas eskalasi. Service desk bertanggung jawab mengawal setiap progress penanganan, menginformasikannya kepada pengguna, hingga menutup laporan.



Gambar 6.6 Langkah-Langkah Incident Management

Aktivitas Incident Management sering berinteraksi dengan proses-proses manajemen layanan TI lainnya, yaitu:

1. Service Level Management
2. Information Security Management
3. Capacity Management
4. Availability Management
5. Service Asset and Configuration Management
6. Change Management
7. Problem Management
8. Access Management

Problem Management

Problem adalah akar penyebab satu atau lebih dari incident. **Problem Management** adalah proses menganalisis dan menyelesaikan akar penyebab incident.

Tujuan dari proses Problem Management adalah mengelola siklus sebuah masalah, mulai dari identifikasi akar masalah, investigasi lanjut, dokumentasi, hingga penyelesaian masalah, termasuk:
1. Menyelesaikan akar penyebab masalah dan incident yang diakibatkannya terjadi
2. Tindakan proaktif mencegah incident yang pernah terjadi terulang di masa depan
3. Meminimalkan dampak buruk dari incident yang tidak dapat dicegah, yang umumnya disebabkan oleh kerusakan (errors) di infrastruktur TI

Perbedaan antara Problem Management dengan Incident Management adalah bahwa **Problem Management** menggunakan teknik-teknik untuk menyelesaikan akar penyebab sebuah "gejala" masalah, sedangkan **Incident Management** menggunakan teknik-teknik untuk menyelesaikan "gejala" masalah (symptoms) saja.

Contoh

Analogi dari incident (gejala) dan problem (akar masalah) adalah sakit kepala. Sakit kepala adalah gejala, kejadian yang menginterupsi kondisi tubuh kita, sehingga mirip dengan incident. Sedangkan sakit kepala ini kemungkinan merupakan gejala dari berbagai kemungkinan penyebab penyakit, misalnya tekanan darah tinggi, kolesterol tinggi, kanker otak, dan lain-lain. Dengan meminum obat sakit kepala hanyalah akan menghilangkan sementara gejala penyakitnya, bukan sumber penyakitnya, sehingga kemungkinan sakit kepala akan datang kembali di kemudian hari.

Cakupan Problem Management meliputi:
1. Aktivitas-aktivitas yang diperlukan untuk mendiagnosis akar penyebab masalah incident dan merumuskan penyelesaiannya
2. Bertanggung jawab memastikan solusi akar masalah tersebut benar-benar diimplementasikan dengan prosedur kontrol yang benar, khususnya melalui proses Change Management dan Release and Deployment Management
3. Menyimpan dan menjaga informasi tentang masalah (problems), workaround, dan penyelesaian tuntasnya (Known Error Database/KEDB)

Reactive vs. Proactive Problem Management

Terdapat dua jenis proses Problem Management, yaitu:
1. Reactive Problem Management adalah tindakan mencari akar permasalahan dan menyelesaikannya dipicu oleh sebuah/lebih kejadian (atau sebagai reaksi dari) incident
2. Proactive Problem Management adalah tindakan mencari dan menyelesaikan akar sebuah masalah tanpa menunggu incident terjadi, misalnya dengan melihat pola, tren, dan frekuensi incident, sebagai bagian dari aktivitas meningkatkan layanan, khususnya availability dan kepuasan pelanggan terhadap layanan TI

Contoh

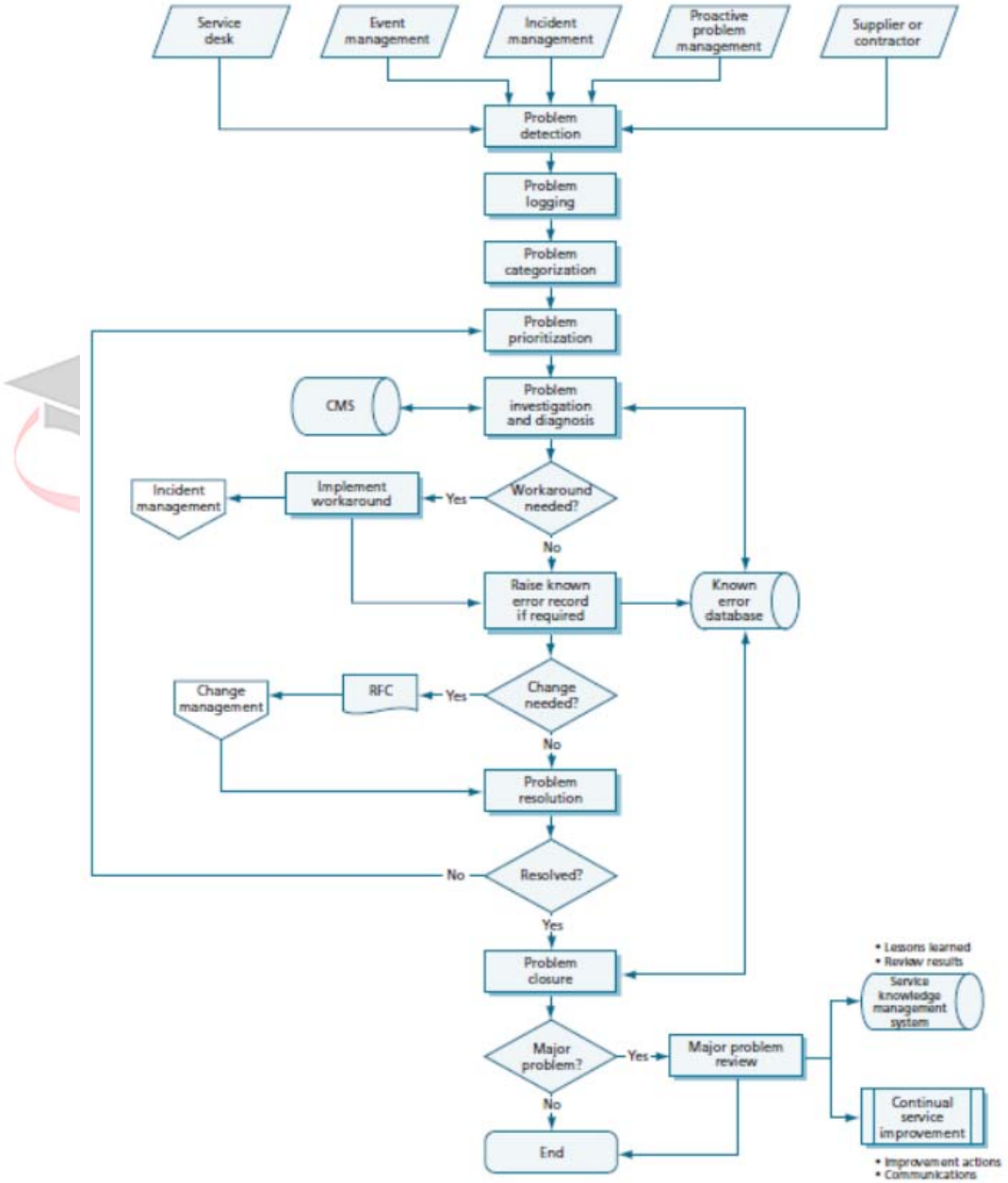
Router sering hang sehingga layanan akses Internet sering terganggu adalah contoh **incident**. Sebagai tindakan **Incident Management**, staf service desk selalu melakukan restart pada router tersebut sebagai tindakan standar solusi sementara (**workaround**) untuk mengembalikan layanan Internet segera pulih kembali. Setiap bulan sekali, manajemen TI selalu mengadakan pertemuan untuk membahas incident yang terjadi dan menganalisis akar masalahnya (**Root Cause Analysis** atau **RCA**). Untuk kejadian router sering hang tersebut diketahui akar penyebabnya adalah suhu yang terlalu tinggi pada router (overheat), hal ini disebut **Known Error**. Data hasil analisis incident dan akar penyebabnya ini selanjutnya disimpan dalam **Known Error Database** (**KEDB**) dan sebagai tindakan lanjut penyelesaiannya, misalnya penggantian semua router dengan jenis terbaru yang memiliki kinerja lebih lama dan daya tahan panas yang lebih baik, diajukan dalam bentuk **Request for Change** (**RFC**) yang merupakan aktivitas dalam proses Change Management. RFC ini disampaikan di rapat **Change Advisory Board** (**CAB**), yang umumnya diadakan secara rutin pada periode waktu tertentu, untuk memperoleh persetujuan hingga masalah terselesaikan. Semua produk dari Problem Management, baik reactive maupun proactive Problem Management, membutuhkan persetujuan CAB.

Aktivitas-Aktivitas dalam Problem Management

Aktivitas-aktivitas dalam Problem Management mirip dengan aktivitas dalam Accident Management. Aktivitas Problem Management umumnya dilakukan periodik/terjadwal (misalnya sebulan sekali atau tiga bulan sekali) dalam bentuk:

1. Pertemuan membahas evaluasi major incident, mencari akar penyebab, dan mengambil langkah-langkah perulangan kejadian
2. Pertemuan mengevaluasi catatan operasional (logs) dan pemeliharaan serta tren event infrastruktur TI
3. Sesi brainstorming untuk mengidentifikasi tren incident, event, dan unjuk kerja sistem TI
4. Aktivitas rutin mengisi lembar checklist untuk secara proaktif mengumpulkan data dan masalah-masalah kualitas operasional layanan TI dalam rangka membantu mendeteksi adanya sebuah masalah

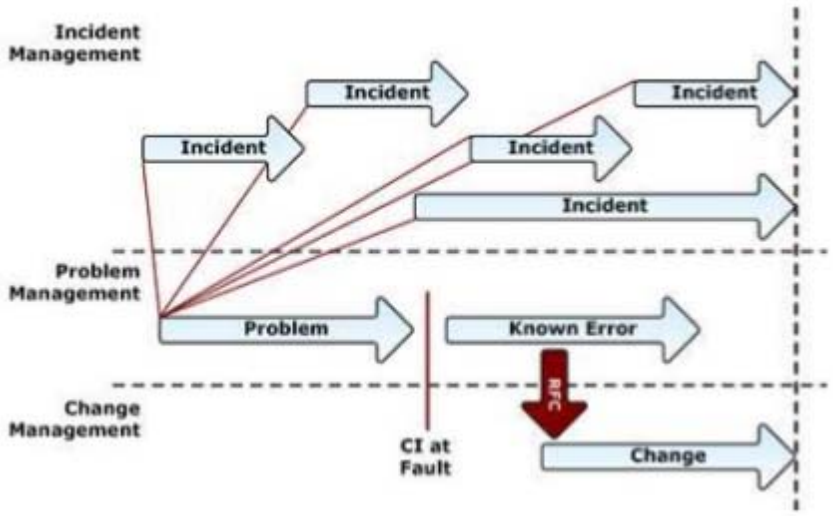
Secara urutan, langkah-langkah Problem Management dapat dilihat pada gambar berikut.



Gambar 6.7 Langkah-Langkah Problem Management

1. **Problem detection and problem logging**
   Proses Problem Management diawali dengan aktivitas mengenali (**detection**) dan mencatat penyebab masalah (problem). Pencatatan sebuah problem (**logging**) dilakukan oleh staf service desk berdasarkan hasil masukan dari berbagai sumber, termasuk service desk sendiri, Event Management, Incident Management, proactive Problem Management, atau supplier. Kriteria incident atau sesuatu dapat dianggap sebagai problem dapat ditentukan oleh organisasi sendiri, termasuk di antaranya:
   a. Saat service desk menyelesaikan sebuah incident, mereka menemukan penyebab incident perlu diinvestigasi lebih lanjut sebagai catatan sebuah problem
   b. Hasil analisis dari tim teknis pendukung Incident Management
   c. Pemberitahuan dari supplier bahwa ada sebuah problem (misalnya bug dari sebuah software) yang masih belum terpecahkan
   d. Hasil analisis rutin tren incident yang terjadi oleh tim Problem Management sebagai bagian dari proses proactive Problem Management
   e. Otomatisasi software service desk, misalnya diatur jika incident dengan prioritas tertentu terjadi minimal 5 (lima) kali sehari, maka software akan otomatis membuat catatan sebuah problem
2. **Problem categorization and prioritization**
   Dalam proses pencatatan, sebuah problem juga perlu dikategorisasi dan ditentukan tingkat prioritasnya dengan metode yang sama dengan metode pengkategorisasian dan prioritas incident (lihat aktivitas incident categorization dan incident prioritization).
3. **Problem investigation and diagnosis**
   Ini adalah tahapan ketika tim teknis, atau dapat juga bekerja sama dengan supplier, melakukan sebuah investigasi menemukan akar permasalahan layanan TI tertentu dan pilihan-pilihan solusinya. Aktivitas ini seringkali membutuhkan akses ke Configuration Management System (CMS) dan Known Error Database (KEDB). Umumnya aktivitas ini dijalankan dalam bentuk proyek kecil.
   Seringkali dalam penyelesaian sebuah problem belum dapat ditemukan solusi tuntasnya, sehingga untuk mengurangi dampak buruknya dibutuhkan penyelesaian sementara atau **workaround**. Workaround selanjutnya diimplementasikan dalam proses Incident Management.
4. Mencatat sebuah **known error**
   Apabila akar permasalahan telah ditemukan dan diselesaikan dalam bentuk workaround, maka catatan detil tentang problem tersebut (gejalanya, incident terkait, solusi standarnya atau workaround, dan lain-lain) atau disebut known error segera dibuat dan disimpan di Known Error Database (KEDB).
5. **Problem resolution**
   Berdasarkan hasil problem investigation and diagnosis, selanjutnya dipilih solusi atau workaround yang diaplikasikan untuk menyelesaikan problem. Apabila ternyata solusi permasalahannya membutuhkan perubahan dalam infrastruktur atau layanan TI, maka sebuah **dokumen RFC harus dibuat** melalui proses Change Management.
6. **Problem closure**
   Ketika sebuah permasalahan telah selesai diatasi dan catatan known error juga telah dibuat, maka catatan problem dapat diberi status "selesai" dan ditutup.
7. **Major problem reviews**
   Khusus untuk permasalahan-permasalahan yang memiliki dampak besar bagi bisnis (major problems), review terhadap penyelesaian permasalahan tersebut perlu dilakukan, termasuk di antaranya mengkaji:
   a. Apa-apa yang telah dilaksanakan dengan benar
   b. Kesalahan-kesalahan sebelumnya
   c. Apa yang masih dapat ditingkatkan
   d. Bagaimana mencegah kejadian serupa
   e. Tindakan lanjut apa yang harus diambil kemudian

Semua hasil review tersebut (termasuk lessons learned) selanjutnya disimpan dalam Service Knowledge Management System (SKMS) dan ditindaklanjuti melalui proses-proses Continual Service Improvement (CSI).



Gambar 6.8 Hubungan Antara Incident, Problem, dan Change

Request Fulfilment

Permintaan layanan atau **service request** adalah permintaan pengguna tentang informasi tertentu, pertanyaan atau permintaan saran, perubahan yang bersifat standar (standard change), atau akses ke suatu layanan TI. Permintaan layanan ini umumnya ditangani oleh service desk tanpa perlu membuat/mengirim RFC. Service request mencakup:

1. Pertanyaan atau permintaan informasi (**request for information**)

   Contoh

   Seorang pengguna bertanya bagaimana cara mencetak dokumen dengan printer yang disediakan untuk publik.

2. Perubahan standar (**standard change**)

   Contoh

   Permintaan me-reset password atau menginstalasi aplikasi tertentu.

3. Pujian atau keluhan (**compliments** or **complaints**)

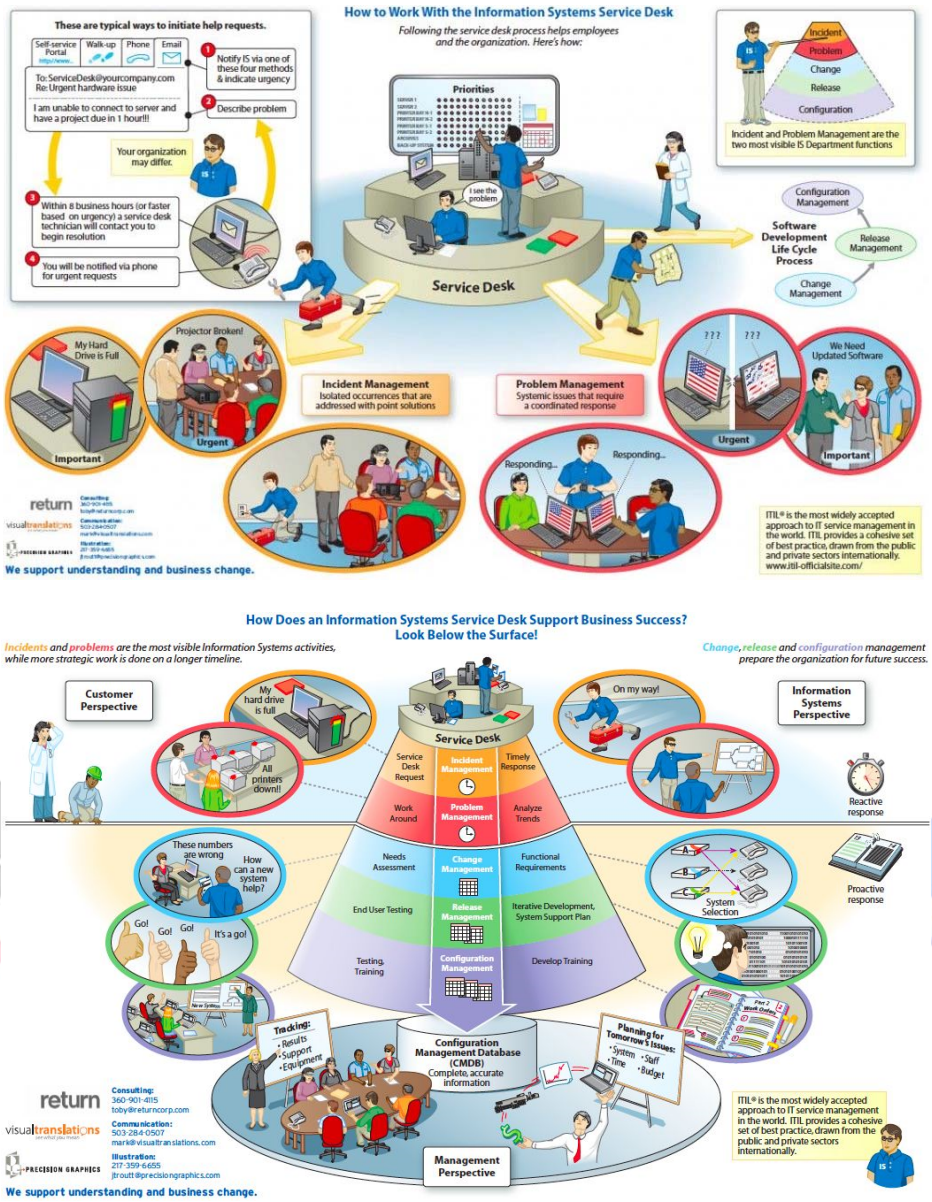
   Contoh

   Seorang pengguna menyampaikan keluhan kecepatan akses internet lebih lambat dari biasanya.

Request Fulfilment adalah proses memenuhi permintaan pengguna layanan TI, di luar laporan terkait dengan incident TI. Karena service desk adalah pusat layanan bagi pelanggan (single point of contact) untuk semua hal terkait layanan TI, maka umumnya service desk akan menerima berbagai macam bentuk laporan, panggilan, atau permintaan, dari pelaporan incident, saran, pertanyaan, hingga permintaan tertentu.

Dalam definisi ITIL, Request Fulfilment adalah sebuah proses yang bertanggung jawab untuk mengelola siklus hidup semua permintaan layanan TI dari pengguna.

Pada dasarnya, service desk bertugas memilah panggilan/laporan yang masuk, apakah sebagai sebuah laporan incident, sebuah permintaan akses, atau sebuah permintaan lainnya di luar incident dan akses. Selanjutnya, semua panggilan/laporan tersebut dicoba untuk diselesaikan oleh service desk sesuai dengan SOP proses masing-masing (Incident Management, Access Management, atau Request Fulfilment). Apabila membutuhkan dukungan dari fungsi lain, maka diteruskan ke pihak terkait (IT Operation teams dan Technical atau/dan Application teams).



Gambar 6.9 Service Desk dan Prosesnya

Tujuan dari proses Request Fulfilment adalah:

1. Memenuhi dan menjaga kepuasan pelanggan dengan menindaklanjuti setiap permintaan layanan secara efisien dan profesional
2. Menyediakan saluran bagi pengguna untuk menyampaikan permintaan layanan dan menerima layanan standar
3. Menyediakan informasi bagi pengguna dan pelanggan tentang layanan-layanan TI yang tersedia dan prosedur untuk mengaksesnya
4. Menyediakan sumber daya layanan TI untuk layanan standar (misalnya software berlisensi resmi)
5. Menyediakan bantuan informasi, menerima dan menindaklanjuti keluhan dan komentar

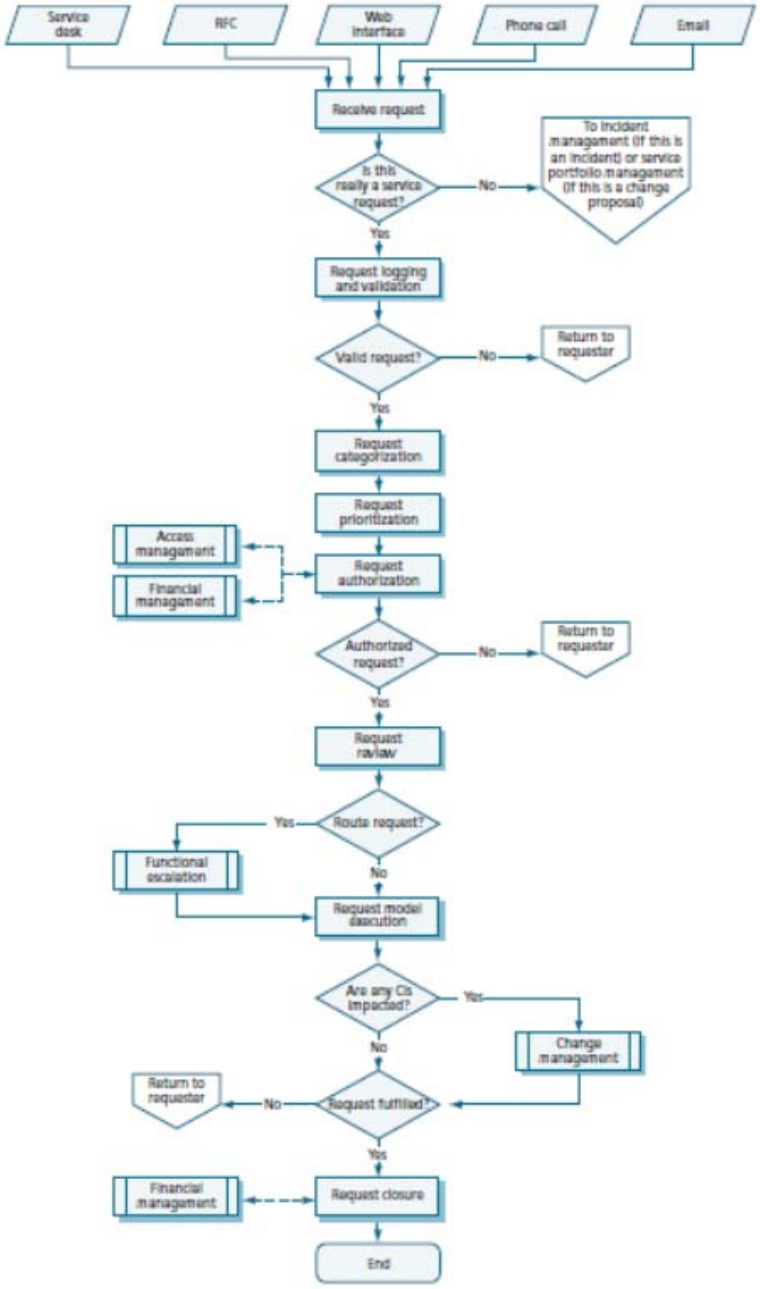
Cakupan proses Request Fulfilment meliputi layanan-layanan yang telah ditentukan oleh setiap organisasi yang dapat ditangani melalui proses Request Fulfilment dan mana yang harus melalui proses lain, seperti Business Relationship Management untuk permintaan layanan-layanan baru atau termodifikasi.

Istilah-Istilah dalam Request Fulfilment

Beberapa istilah dalam pemenuhan permintaan:

1. **Request model**
   Standar prosedur (SOP) untuk menangani tipe-tipe permintaan tertentu yang rutin terjadi.
   Contoh
   Permintaan pendaftaran peralatan pribadi untuk mengakses sistem (Bring Your Own Device atau BYOD).
2. **Self-help technology**
   Teknologi berbasis web yang memungkinkan pengguna memilih dan mencatat sendiri permintaan mereka. Umumnya dalam bentuk formulir online dan pengguna diharuskan melakukan login ke sistem dengan user name dan password tertentu untuk memastikan identitas pengguna.

Aktivitas-Aktivitas dalam Request Fulfilment



Gambar 6.10 Aktivitas dalam Request Fulfilment

Aktivitas-aktivitas proses Request Fulfilment adalah:

1. **Receive request, request logging, and validation**
   Aktivitas menerima, mencatat, dan memastikan/menvalidasi sebuah permintaan layanan. Penerimaan permintaan layanan dapat dilakukan melalui formulir kertas, e-mail, RFC, atau menggunakan self-help system yang memungkinkan pelanggan untuk mencatat permintaannya sendiri. Setiap permintaan diterima oleh service desk, kemudian dicatat detil permintaannya dan divalidasi. Catatan permintaan (request record) juga harus mencatat detil hasil pemenuhannya.
2. **Request categorization, request prioritization**
   Seperti dalam Incident Management dan Problem Management, setiap permintaan (request) yang masuk juga harus dikategorisasi dan ditentukan skala prioritas penanganannya berdasarkan pertimbangan dampak dan urgensinya. Waktu pemenuhan permintaan layanan juga harus sesuai standar layanan yang telah disepakati dalam SLA, serta dikomunikasikan kepada staf service desk dan pelanggan.
3. **Request authorization**
   Beberapa permintaan dapat langsung dipenuhi oleh staf service desk karena kewenangan memang di tangan staf service desk, namun beberapa permintaan mungkin menyangkut kewenangan pihak lain, misalnya apabila permintaan menyangkut biaya atau perubahan infrastruktur layanan TI. Aktivitas ini adalah aktivitas memastikan orang/pihak yang berwenang menyetujui pemenuhan layanan.
4. **Request review, request model execution**
   Request model adalah prosedur standar (SOP) pemenuhan permintaan-permintaan tertentu. Pada aktivitas ini, staf service desk me-review permintaan untuk menentukan request model mana yang paling sesuai, selanjutnya melaksanakannya.
5. **Request closure**
   Service desk bertanggung jawab untuk memastikan permintaan telah dipenuhi sesuai harapan pelanggan. Hati-hati jika pemberitahuan pemenuhan permintaan dikirimkan via e-mail, karena harus dipastikan e-mail sampai ke pengguna dan pengguna dapat merespon kepada service desk jika masih ada permasalahan lain.

Access Management

Access didefinisikan sebagai fungsi dan data layanan apa saja yang dapat digunakan oleh pengguna. Jika dianalogikan dengan rumah, maka access adalah ketentuan ruangan-ruangan mana saja yang dapat dibuka dan dimasuki.

Access Management adalah proses pengelolaan hak akses pengguna ke sistem layanan TI. Tiap organisasi harus mempunyai kebijakan yang mengatur "Siapa dapat mengakses layanan TI apa?" dan bagaimana cara orang-orang tersebut meminta akses layanan. Kebijakan ini telah ditetapkan dalam proses Information Security Management dan Availability Management di tahapan Service Design. Jadi Access Management adalah implementasi dari proses Information Security Management dan Availability Management dalam operasional layanan TI sehari-hari. Proses Access Management seringkali disebut sebagai **Rights Management** atau **Identity Management**.

Tujuan dari proses Access Management adalah:

1. Menyediakan hak akses bagi pengguna yang berhak (authorized user) untuk dapat menggunakan satu layanan atau grup layanan TI
2. Mencegah pengguna yang tidak berhak (unauthorized user) untuk mengakses layanan TI

Cakupan proses Access Management meliputi:

1. Implementasi dari kebijakan-kebijakan yang telah ditetapkan dalam Information Security Management, dalam rangka menjaga kerahasiaan (confidentiality), ketersediaan (availability), serta kebenaran (integrity) data dan properti intelektual organisasi
2. Memastikan pengguna memperoleh hak akses sebuah layanan, tetapi tidak memastikan ketersediaan layanan setiap saat (ini ditangani proses Availability Management)
3. Aktivitas-aktivitas proses Acccess Management umumnya dilaksanakan oleh fungsi Technical Management (jika menyangkut hardware) dan Application Management (jika menyangkut software), namun umumnya dikoordinasi di satu titik kontak, yaitu IT Operations Management atau di service desk

4. Access Management dapat diinisiasi oleh sebuah permintaan layanan (proses Request Fulfilment)

Istilah-Istilah dalam Access Management

Beberapa istilah dalam Access Management:

1. **Access request**
   Permintaan akses layanan, dapat berupa permintaan layanan standar atau RFC.
2. **Information security policy**
   Peraturan/kebijakan implementasi Access Management (siapa dapat mengakses sistem dan/atau data apa, dapat melakukan apa saja) dibuat di proses Information Security Management.
3. **Identitas (identity)**
   Informasi unik setiap pengguna, yang menjelaskan status pengguna di dalam organisasi dan sekaligus menentukan hak aksesnya terhadap sistem layanan TI.
4. **Rights** atau **privileges**
   Hak akses seorang/grup pengguna ke satu/kelompok layanan TI, misalnya hanya dapat membaca file (read), atau dapat menghapus (delete), atau dapat mengubah data (edit). Jika dianalogikan dengan rumah, maka mengatur ketentuan kegiatan-kegiatan apa saja yang dapat dilakukan setelah diperbolehkan memasuki ruangan tertentu.
5. **Services** atau **service groups**
   Sekumpulan layanan yang dibutuhkan user untuk melakukan suatu aktivitas bisnis yang mirip. Jika dianalogikan dengan rumah, maka untuk seorang pengguna dengan tugas tertentu diberikan 1 set ring yang berisi kunci-kunci ruangan terkait yang dapat dia buka dan masuki.

   Contoh

   Jika seorang pengguna telah didaftarkan di sistem sebagai anggota grup departemen penjualan, maka ia akan secara otomatis memperoleh hak akses ke sistem customer relationship management, sistem pemesanan penjualan, dan sistem informasi tagihan.
6. **Directory services**
   Alat atau software yang dipergunakan oleh service desk atau tim pendukung layanan TI untuk mengatur manajemen akses.

Aktivitas-Aktivitas dalam Access Management

Secara teknis, Access Management adalah sekumpulan aktivitas mendefinisikan profil pengguna dan password untuk setiap sistem layanan TI yang dapat digunakan. Access Management juga dilaksanakan oleh service desk. Proses Access Management dilakukan melalui aktivitas-aktivitas:

1. **Permintaan akses (access request)**
   Aktivitas proses manajemen akses dimulai dengan permintaan akses (access request) yang dapat disampaikan dalam bentuk formulir permintaan layanan dari proses Request Fulfilment atau RFC.

   Contoh

   Kasus layanan TI baru untuk 1000 pengguna.
2. **Verifikasi (verification)**
   Aktivitas memverifikasi identitas pengguna yang mengajukan permintaan akses.

   Contoh

   Dengan mengirimkan link konfirmasi ke e-mail resmi pengguna, menelepon pengguna, atau menanyakan beberapa pertanyaan rahasia (seperti tempat tanggal lahir, nama ibu kandung, dan lain-lain).
3. **Menyediakan hak akses (providing rights)**
   Setelah permintaan akses dan identitas pengguna divalidasi, selanjutnya system administrator atau network administrator pada departemen TI menyediakan hak akses ke layanan TI.
4. **Memonitor status identitas, mencabut/membatasi hak akses (monitoring identity status, removing or restricting rights)**
   Aktivitas memantau perubahan status identitas pengguna, dan mengubah hak akses sesuai status terkini. Untuk memastikan status terkini seorang pengguna, umumnya sistem Access Management terkoneksi atau bekerja sama dengan sistem departemen sumber daya manusia.

   Contoh

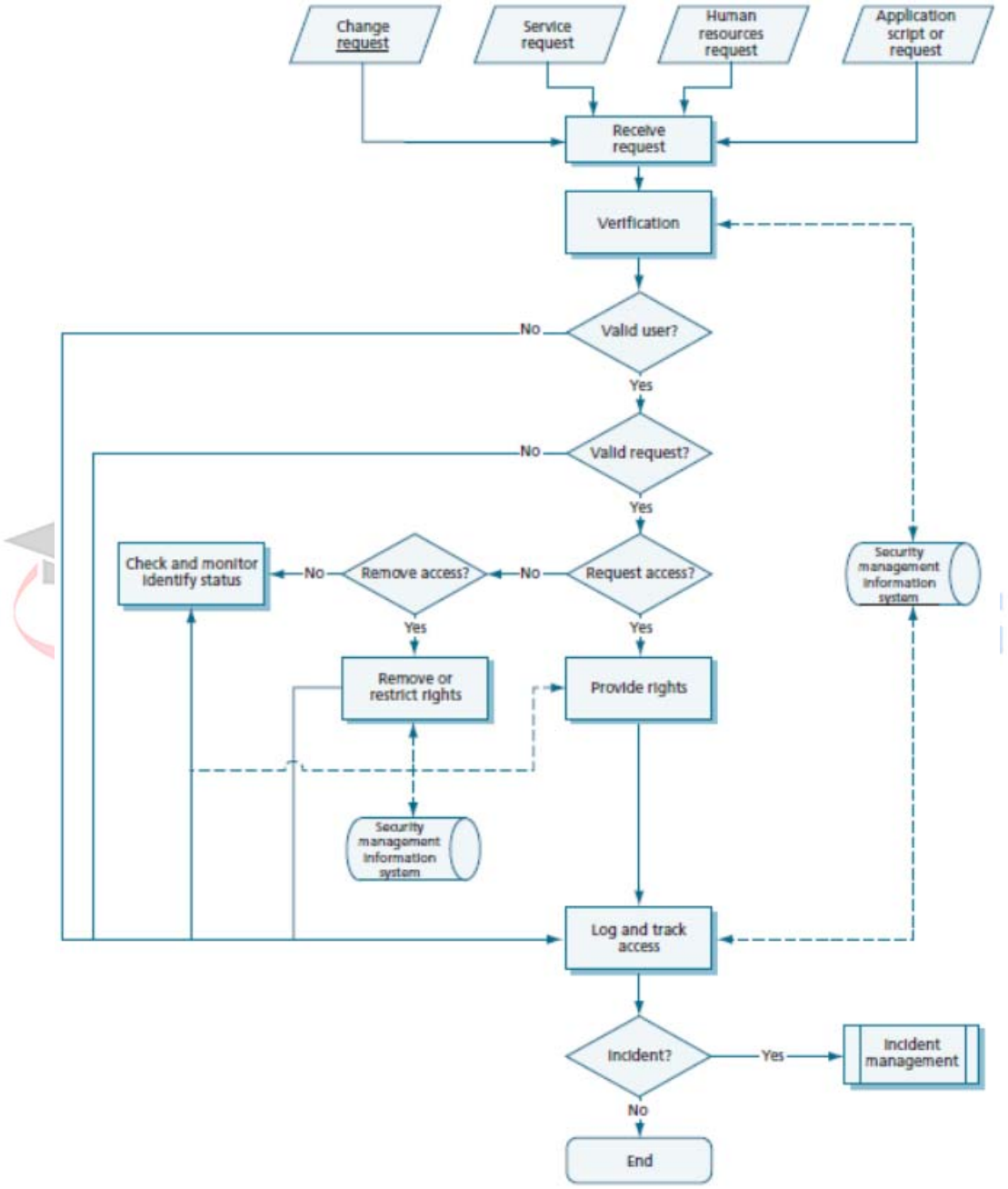
   Ketika pengguna tidak lagi bekerja di organisasi tersebut, maka hak akses pengguna tersebut dicabut dalam kurun waktu yang ditentukan, atau apabila pengguna berganti peran dalam organisasi, maka hak aksesnya ditambah atau dikurangi sebagaimana mestinya.

5. Pencatatan dan penelusuran aktivitas akses (logging and tracking access)

Aktivitas pencatatan dan pelacakan akses yang mencurigakan. Biasanya hal ini dilakukan otomatis dengan monitoring tools. Semua aktivitas yang tidak biasa harus dicatat sebagai sebuah incident dan diinvestigasi lebih lanjut.

Contoh

Percobaan akses layanan secara ilegal, atau jumlah kesalahan input password yang berlebihan.



Gambar 6.11 Aktivitas dalam Access Management

Peran-Peran (Roles) dalam Service Operation

Proses-proses dalam Service Operation membutuhkan beberapa peran (roles) sebagai berikut:

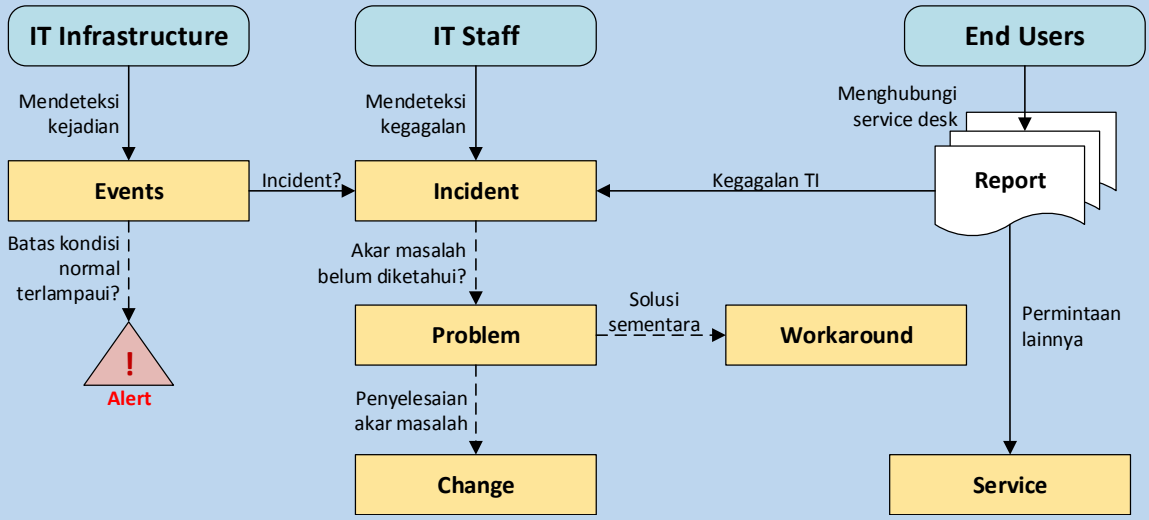
1. **1st Level Support (Service Desk Analyst)**
   Bertanggung jawab menerima, mencatat, dan mengklasifikasikan laporan incident dan mengambil tindakan pertama dalam mengembalikan layanan yang terinterupsi secepat mungkin, serta melayani permintaan layanan TI (service request) yang masuk (termasuk dalam proses Incident Management, Request Fulfiment, dan Access Management), dan memastikan setiap pengguna menerima informasi perkembangan (status) penanganan incident sesuai standar layanan. Apabila tidak dapat memberikan solusi, maka 1st level support akan meneruskan penyelesaian incident ke technical support groups (2nd level support).
2. **2nd Level Support**
   Peran yang bertanggung jawab mengambil alih tindakan penyelesaian incident yang tidak mampu diselesaikan oleh 1st level support. Apabila dibutuhkan, 2nd level support dapat meminta bantuan pihak eksternal, seperti manufaktur software atau hardware. Apabila solusi incident masih saja belum mampu ditemukan, maka 2nd level support meneruskan permasalahan incident ke proses Problem Management.
3. **3rd Level Support**
   Peran pendukung layanan operasional yang umumnya dilakukan oleh pengembang hardware atau software. Layanan mereka diminta oleh 2nd level support saat dibutuhkan untuk penyelesaian sebuah incident.
4. **Incident Manager**
   Bertanggung jawab melaksanakan proses Incident Management yang efektif dan prosedur laporan (jenjang pertama eskalasi vertikal incidents yang tidak dapat diselesaikan sesuai standar layanan).
5. **Problem Manager**
   Bertanggung jawab mengelola siklus hidup semua masalah layanan TI (problems). Tanggung jawab utamanya adalah mencegah incident yang sama terulang kembali dan meminimalkan dampak dari incident yang tidak dapat dicegah. Oleh karena itu, problem manager bertugas menjaga keterbaruan informasi tentang known errors dan workarounds (known error database).
6. **Service Desk Manager**
   Bertanggung jawab mengatur staf service desk, mengatasi insiden dan permintaan layanan pada service desk, serta proses eskalasi kasus-kasus yang sulit. Seorang service desk manager dapat juga menjalankan peran incident manager dan problem manager sekaligus.
7. **Major Incident Team**
   Peran yang dilakukan sebuah tim yang berisi manajer-manajer TI dan ahli teknis TI yang umumnya berada di bawah koordinasi incident manager, yang dibentuk khusus untuk penyelesaian masalah-masalah besar incident (major incident).
8. **Access Manager**
   Bertanggung jawab memberikan hak akses yang tepat kepada setiap pengguna untuk sebuah layanan TI dan mencegah orang-orang yang tidak berhak mengakses layanan TI. Secara umum, seorang access manager melaksanakan kebijakan-kebijakan yang sudah ditetapkan di proses Information Security Management.
9. **IT Operations Manager**
   Bertanggung jawab mengambil berbagai tindakan atau aktivitas IT Operations Management, termasuk di dalamnya Operations Control dan Facilities Management.
10. **IT Operator**
    Staf yang bertanggung jawab melaksanakan aktivitas operasional harian IT Operations Management, seperti melakukan back-up, perawatan rutin, instalasi peralatan standar data center, dan lain-lain.
11. **IT Facilities Manager**
    Bertanggung jawab mengelola lingkungan fisik sistem TI, termasuk listrik, sistem pendingin, manajemen akses gedung, monitoring keamanan lingkungan, dan lain-lain.

Teknologi Penunjang (Tools) Service Operation

Kebanyakan aktivitas dalam Service Operation adalah kegiatan rutin dan berulang sehingga lebih efisien jika diotomatisasi dengan memanfaatkan teknologi pendukung, seperti service desk logging tools atau integrated service management toolset. Apapun nama software pendukung tersebut, disarankan menyediakan beberapa fungsi sebagai berikut:

1. **Self-help**
   Kemampuan pengguna untuk mengakses informasi layanan, log-in ke sistem, mencatat dan mengirimkan permintaan atau incident secara mandiri kepada penyedia layanan.
2. **Workflow** atau **process engine**
   Kemampuan otomatisasi proses dengan meneruskan incident, service request, dan perubahan (change) ke staf yang berwenang.
3. **Integrated CMS (Configuration Management System)**
   Memiliki alat dan basis data yang berisi informasi layanan-layanan TI yang tersedia, aset dan komponen yang mendukungnya, serta informasi hubungan antar aset dengan incident, problem, known error, dan change records terkait.
4. **Discovery/deployment/licensing technology**
   Discovery tools berfungsi secara otomatis menemukan dan mengenali komponen-komponen dalam infrastruktur sistem TI dan mengumpulkan informasi tersebut. Deployment tools berfungsi mendistribusikan/menginstalasi software ke target-target PC secara otomatis. Licencing tools berfungsi mencatat, memeriksa, dan memonitor lisensi dari setiap software yang terinstalasi.
   a. **Remote control**
      Kemampuan mengambil alih PC/workstation pengguna, misalnya untuk keperluan menginstalasi software atau mendiagnosis masalah di PC tersebut.
   b. **Diagnostic tools**
      Tools untuk membantu menemukan dan mendiagnosis masalah.
   c. **Reporting and dashboards**
      Kemampuan untuk membuat laporan serta menampilkan statistik dan pengukuran kinerja TI.

Berikut ini beberapa rangkuman konsep-konsep dalam Service Operation.



Gambar 6.12 Konsep-Konsep dalam Service Operation