Table of Contents

1. About Tech Mahindra

1.1 Overview

Tech Mahindra is part of the US $15.4 billion Mahindra Group and is a leading global systems integrator and business transformation consulting organization, focused primarily on the telecommunications industry. Tech Mahindra expanded its IT portfolio in 2009 by acquiring the leading global business and information technology services company, Mahindra Satyam(earlier known as Satyam Computer Services).

Tech Mahindra is ranked #6 in India's software services firms behind Tata Consultancy Services, Wipro, Infosys, HCL Technologies and Satyam Computer Services and overall #161 in Fortune India 500 list for 2011.

1.2 History

Tech Mahindra was incorporated as a joint venture between Mahindra & Mahindra and British Telecom plc in 1986 under the name of ‘Mahindra British Telecom’. Later, the name was changed to ‘Tech Mahindra’, to reflect the diversification and growth of the client base and the increased breadth of our service offerings.

1.3 Milestones

- **1986** - Incorporation in India.
- **1987** - Commencement of business.
- **1993** - Incorporation of MBT International Inc. The first overseas subsidiary.
- **1995** – Established the UK branch office.
- **2001** – Incorporated MBT GmbH, Germany incorporated.
- **2003** - Incorporated MBT Software technologies ,Singapore.
- **2005** – Acquired Axes Technologies Ltd, including its US and Singapore Offices
- **2006** – Raised Rs 4.65 Billion ($100 million) from a hugely successful IPO to build a new facility in pune, to house about 9000 staff.
- **2007** – Acquired iPolicy private Ltd.

- **2009 –**With acquisition of Satyam, Tech Mahindra is well positioned to be aleader in the broader IT services space.
- **2010** – AT&T 2010 Supplier award for outstanding performance and service to AT&T and its affiliates.

1.4 Core Purpose

We will challenge conventional thinking and innovatively use all our resources to drive positive change in the lives of our stakeholders and communities across the world, to enable them to rise.

Values

Tech Mahindra is focused on creating sustainable value growth through innovative solutions and unique partnerships. Our values are at the heart of our business reputation and are essential to our continued success. We foster an environment to instill these values in every facet of our organization.

- Customer first
- Good corporate citizenship
- Professionalism
- Commitment to quality
- Dignity of the individual

1.5 Brand Promise

Tech Mahindra's brand positioning highlights the success of the Company which has emerged as the fastest growing provider of IT Solutions & Services in the Telecom space. The positioning exemplifies the performance of Tech Mahindra, as a success system, which is powering the growth of its stakeholders.

We are a part of the exhilarating Telecom technology space that is characterized by highly innovative, speed-of-response based solutions, delivered with agility and flexibility and our brand reflects this true character. The positioning incorporates both the success of the past and the promise of the future, where we are going to 'Create the next wave' - our brand mnemonic.

2. Introduction

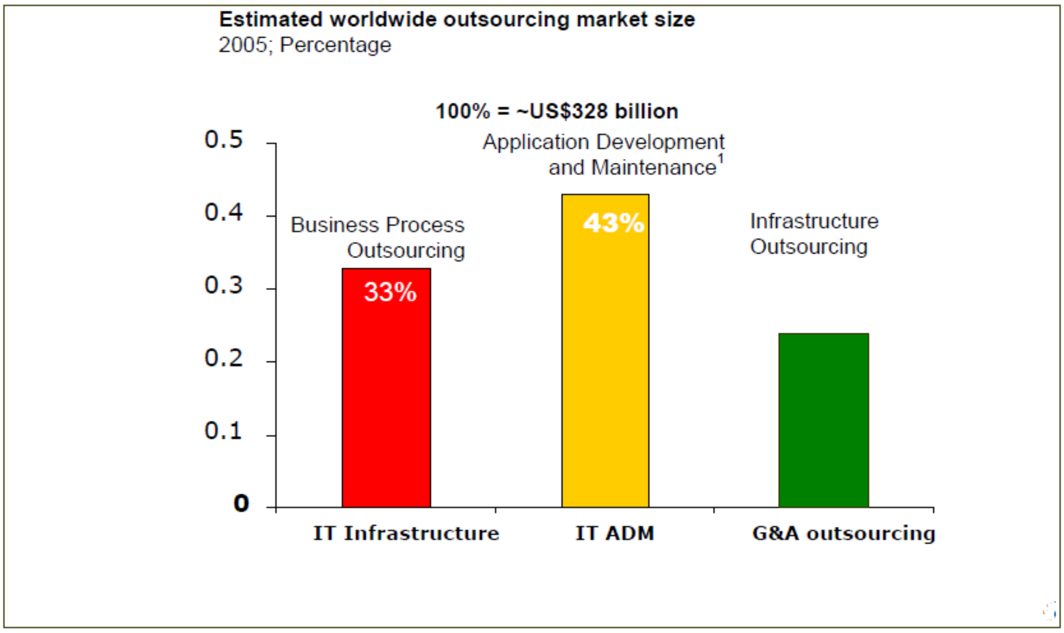
The aim is to understand a new trend in the field of IT Infrastructure.Management called Remote IT Infrastructure Management, which usually caters to the needs of large organisations having a Global Presence, by managing their IT Infrastructure remotely.

Infrastructure Management Services encompasses a broad range of services. It refers to monitoring and management which includes maintenance, administration, troubleshooting, and performance enhancement of the IT Infrastructure of any organization. It covers the entire spectrum of the IT Infrastructure including helpdesk, End User Desktops, sever and storage, databases, telecoms, Network, Security, Application Operations et al.

Infrastructure Management Services is not a new phenomenon, for several years, companies have been working hard to reduce the cost of operating the IT Infrastructure – Datacenters, networks, databases, and business support software tools.

Outsourcing to India can be categorized in the following manner:

Infrastructure outsourcing representing almost 1/3rd of the total outsourcing market.



3. Remote infrastructure management

Remote Infrastructure Management services consist of remote (outside the physical premises of a company's facilities) monitoring and managing the infrastructure components and taking proactive steps and remedial actions across the IT landscape. The remote monitoring and management is undertaken through a combination of offshore/near shore/global delivery center which is often termed as an Offshore Operations Center (OOC), where skilled staff of a service provider monitor and manage the infrastructure, ensuring uptimes and availability.

Some of Infrastructure management services that can be delivered remotely are Network Management Services, Datacenter Outsourcing, Helpdesk outsourcing, Desktop Outsourcing, Storage outsourcing, IT Security Outsourcing, LAN outsourcing, WAN outsourcing, Database and Application Monitoring & Management.

3.1: Evolution of RIM

Gartner Inc., a business intelligence firm, had indicated in its research that around 32% of a CIO's IT budget was spent on internal resources to monitor and manage IT. This area, being a significant chunk, was also the area that required immediate decline.

Once IMS outsourcing became a norm, RIM followed as a natural progression, offering an alternate model of delivery. Some of the key factors that have acted as a catalyst for the RIM being accepted worldwide are:

3.1.1: Maturity of the offshoring (Global Delivery) model:

As companies face continued pressure to cut costs, they have started including infrastructure-related opportunities in their offshoring requirements. This also allows the customer to gain from the scale of operation as is evident from the various multi service deals that have been won by India based Service Providers.

3.1.2: Technical feasibility:

The myth that a IMS outsourcing is associated and bundled with the physical assets of the IT department and the Datacenter and therefore precludes offshore delivery has been broken.

It is estimated that between 60-70% of the IT services surrounding the datacenter, specifically the support functions of monitoring and management can be executed remotely due to the improvements in the remote-monitoring and diagnostic capabilities of leading System and Network management tools.

Reasons like emergence of Business Continuity Services, globalization, increased awareness of RIM market and availability of low cost countries is forcing buyers to think of new ways to service users 24x7 across the globe while increasing efficiency and effectiveness. Moreover organisations are opting for RIM due to advanced security technologies to mitigate risk due to offshoring.

3.2: The Existing Scenario

With the advancement in processes and tools and the proven maturity of Global Delivery for IT Infrastructure Services, today RIM is becoming a strategy of choice for CIOs across the globe who are keen to take advantage of the labour arbitrage without compromising on quality and responsiveness of services.

Today, Offshore vendors have started to invest aggressively both in infrastructure talent and highly available network connectivity globally. At the same time, automation tools, more effective processes, and onshore consolidation efforts have made corporate IT departments increasingly comfortable with locating more of their infrastructure support personnel remotely from assets and users.

3.2.1: Remote Infrastructure Management Services

- Desk Management
- Call Management
- Escalation Management
- Change Management
- Hardware/System Maintenance
- Troubleshooting

Other Features That Are A Part Of RIM Are

- System Administration
- Software Installation

- Configuration And Maintenance
- Backups/Restores
- Network Monitoring And Management
- Basic Reporting
- Advanced Services And Intrusion Detection Service.
- Database Management
- Performance Analysis
- Data Back-up Management

3.3 Why Remote Infrastructure Management:

1. Remote Infrastructure Management Services (RIM) is evolving at quick pace. According to IDC, in a study, estimated that more than 85% of infrastructure components can be managed from remote locations.
2. Service providers are doing big ticket investments to develop their delivery capabilities.
3. In last 2-3 years large clients had delivered remote infrastructure delivery service as a requirement in the majority of new outsourcing deals.
4. Infrastructure services delivered in a global delivery model from India to US companies has surpassed $1 billion.
5. RIM presently contains high risk for global delivery of services. But clients are relying on service providers to help manage and insulate them from risks.
6. High probability that in next 5 years 50% of the labour hours for global IT infrastructure service delivery will shift from on-site support to remote delivery.
7. Increased globalization of organisation hence need for 24X 7 Services.

8. Low cost geographies like Indian with robust Infrastructure and Human Capital capabilities.

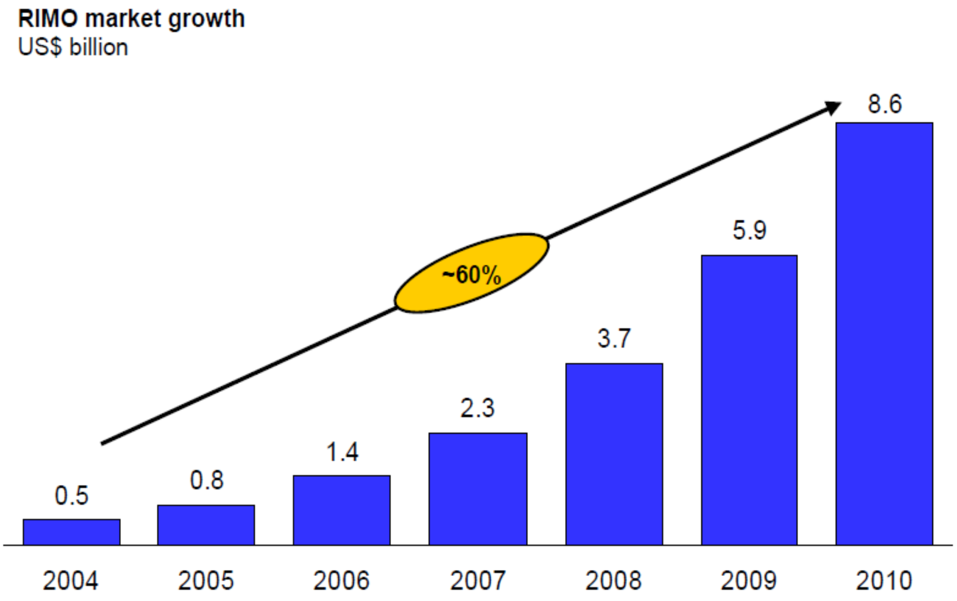
9. Advanced Security Technologies to mitigate risk due to offshoring.

10. Pro-active management by means of enterprise management tools.

11. Need for repeatable, replicable service based on ITIL Model.

    (ITIL: IT Infrastructure Library)

3.4 Growth In RIMS



4. RIM – The India Advantage

Manpower

India has highly skilled labour pool which has ability to cater to global clientele. India has witnessed a healthy growth in the number of its IT professionals. From a base of 6,800 knowledge workers in 1985-86, the number is projected to increase to 650,000 IT software and services professionals by March 2003. It is estimated that out of the 650,000 knowledge workers, almost 205,000 are working in the IT software and services export industry; nearly 160,000 are working in IT enabled services 25,000 in the domestic software market and over 260,000 in user organizations.

Location

Low cost, easy accessibility of India as a location and emerging centers of excellence with multiple cosmopolitan cities is another very significant pull factor for India as a destination of choice.

Infrastructure

The Indian Telecom industry is one of the fastest growing industries .With the entry of the global players there has been tremendous infusion of capital and creation of best of breed infrastructure. India’s capability ,experience and depth in this space has challenged the traditional IT service providers across and range of services.

5. USH – Others LINUX

The server list embedded The ‘USH Non-Solaris - Linux’ is a support team within the ITO organization’s Midrange Technical Systems Support district. Broadly, this team caters to the ITO’s requirement.

TechM has been engaged with AT&T to provide the Tier 1 and 2 support functions.

However, this is not limited to the number of Linux servers or its count at any point of time. Any increase in the count of server will be attributed to the productivity gain of this assignment over the baseline server count that must be decided in agreement of both the parties.

5.1 Work_of system administrators

- Participate in setting up offshore environment.
- Provide Tier-2 Linux System Administration support using AT&T standard monitoring tools, investigate and resolve or coordinate (escalate) to resolve with other teams.
- Handle incoming calls, tickets, Pagers and troubleshoot issues with various groups.
- Participate proactively in conference calls / MMLs or any outage calls.
- Adherence to contractual SLAs and follow AT&T process and procedures when assisting others in resolving issues.
- Communicate SLA status report and critical commitment on daily basis to TechM Project Managers.
- Provide (new/update) technical Documentation as applicable.
- Escalate technical and/or Project issues to Shift Leaders.
- Mentor new hire and provide them training of SA-Linux responsibilities, including process, tool strategies and the network environment.
- Conduct process audits, do root cause analysis to improve productivity and quality.
- Provide (new/update) technical Documentation as applicable.

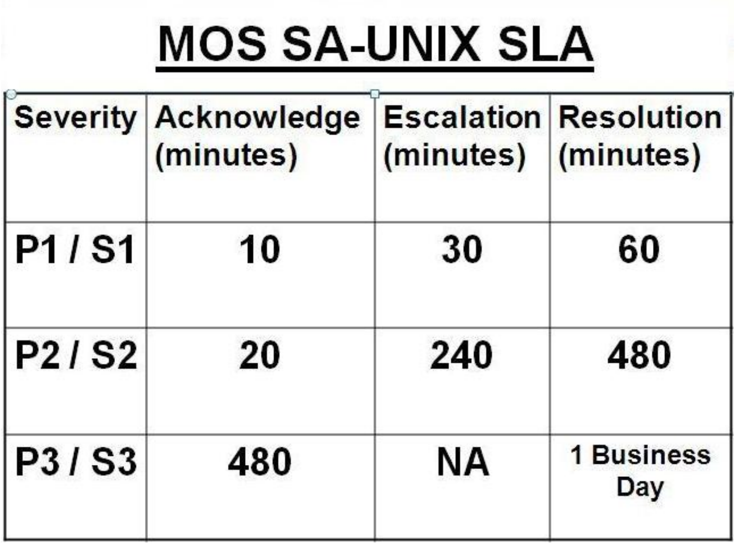
5.2 Service Level Agreement

ITIL Service Level Management aims to negotiate Service Level Agreements with the customers and to design services in accordance with the agreed service level targets. Service

Level Management is also responsible for ensuring that all Operational Level Agreements and Underpinning Contracts are appropriate, and to monitor and report on service levels.

A service level agreement is a document which defines the relationship between two parties: the provider and the recipient. This is clearly an extremely important item of documentation for both parties. If used properly it should

- Identify and define the customer's needs
- Provide a framework for understanding
- Simplify complex issues
- Reduce areas of conflict
- Encourage dialog in the event of disputes
- Eliminate unrealistic expectation



MOS SA-UNIX SLA

Severity	Acknowledge (minutes)	Escalation (minutes)	Resolution (minutes)
P1 / S1	10	30	60
P2 / S2	20	240	480
P3 / S3	480	NA	1 Business Day

- **Severity 1**
  - Tickets should be claimed in 0 – 60 minutes
  - Service should be restored Real Time
  - Repairs should be completed in the next Maintenance window if applicable.
- **Severity 2**
  - Tickets should be claimed in 4 – 8 hours
  - Service should be restored within 48 hours

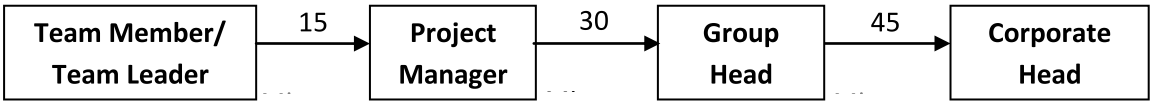
    - Repairs should be completed in the next Maintenance window if applicable.

- **Severity 3 or greater**
    - Tickets should be claimed in 5 business days.

The resource allocation for each team is as below.

Group	Onsite	Offshore	Total Count
USH Non-Solaris Linux	0	25	25

The escalation path used by TechM is:



6. Common tools & technologies

Sr. No	TOOL	DESCRIPTION
1.	IEDS	System information like system owner and O.S platform
2.	Putty	Putty is a free implementation of Telnet and SSH for Win32 and UNIX platforms, along with an xterm terminal emulator.
3.	Amity	Use Amity to connect to the server's console
4.	Q Team-Link Messenger	It is an instant messaging service used by AT&T. With the help of **Q** we can have a chat within the ATT domain and outside the domain using proxy, like any other messaging service.
5.	Live meeting	Using Live Meeting, we can have interactive sessions, give demonstrations, or conduct training sessions.
6.	Remedy	Helpdesk system
7.	XPW	This tool will generate the root password for almost all of the AT&T servers.
8.	Jump Point Server	Jump Point Server is used to login to other servers without issuing any passwords.
9.	Vsphere Client	This tool is used to connect VMware virtual host/guest operating system.
10	VERITAS GUI	VERITAS GUI is used for cluster administration and VERITAS enterprise administrator is used for logical volume management.

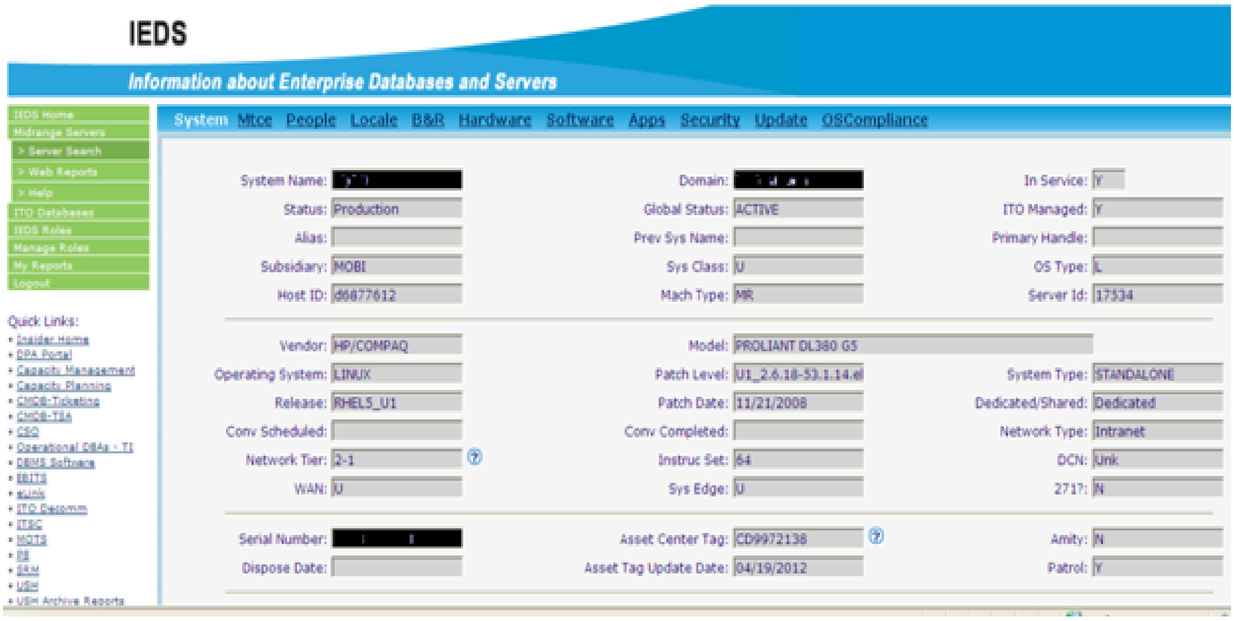
6.1 IEDS (Information about enterprise databases and servers)

IEDS is a centralized information repository of the entire servers; we can get up to 95% correct information regarding the servers, sometimes it may be differing. It is database exclusively used by AT&T for storing all kinds of information about the servers.

Members of Offshore team use this database to find all sorts of information. All the employee got to do is enter the name of the server which he/she gets from the ticket he/she receives.

General information that is always required before solving any ticket is:

1. Check the status of the server i.e whether it production or test or development.
2. Is it in service or expired.
3. Who is the vendor (Dell or HP or any other vendor).
4. What is the system type (standalone or clustered).
5. Is it accessible through amity .
6. Who is the SA team leader . If it is than your SA team leader than it means that this sever is not under our supervision and the ticket has to reverted to the EMOC section.



System Name:	[redacted]	Domain:	[redacted]	In Service:	Y
Status:	Production	Global Status:	ACTIVE	ITO Managed:	Y
Alias:		Prev Sys Name:		Primary Handle:	
Subsidiary:	MOBI	Sys Class:	U	OS Type:	L
Host ID:	d6877612	Mach Type:	MR	Server Id:	17534
Vendor:	HP/COMPAQ	Model:	PROLIANT DL380 G5		
Operating System:	LINUX	Patch Level:	U1_2.6.18-53.1.14.el	System Type:	STANDALONE
Release:	RHEL5_U1	Patch Date:	11/21/2008	Dedicated/Shared:	Dedicated
Conv Scheduled:		Conv Completed:		Network Type:	Intranet
Network Tier:	2-1	Instruc Set:	64	DCN:	Unk
WAN:	U	Sys Edge:	U	271?:	N
Serial Number:	[redacted]	Asset Center Tag:	CD9972138	Amity:	N
Dispose Date:		Asset Tag Update Date:	04/19/2012	Patrol:	Y

This is the interface that opens after one enters the server name.

As one can see the tabs it broadly has information regarding :

- System
- People
- Locale
- B&R
- Hardware
- Software
- Apps (Applications)
- Security

Most important of these above are system ,locale and people as these 3 give us all the information mentioned above that is required before solving the ticket.

6.1.1 System

It contains following information:

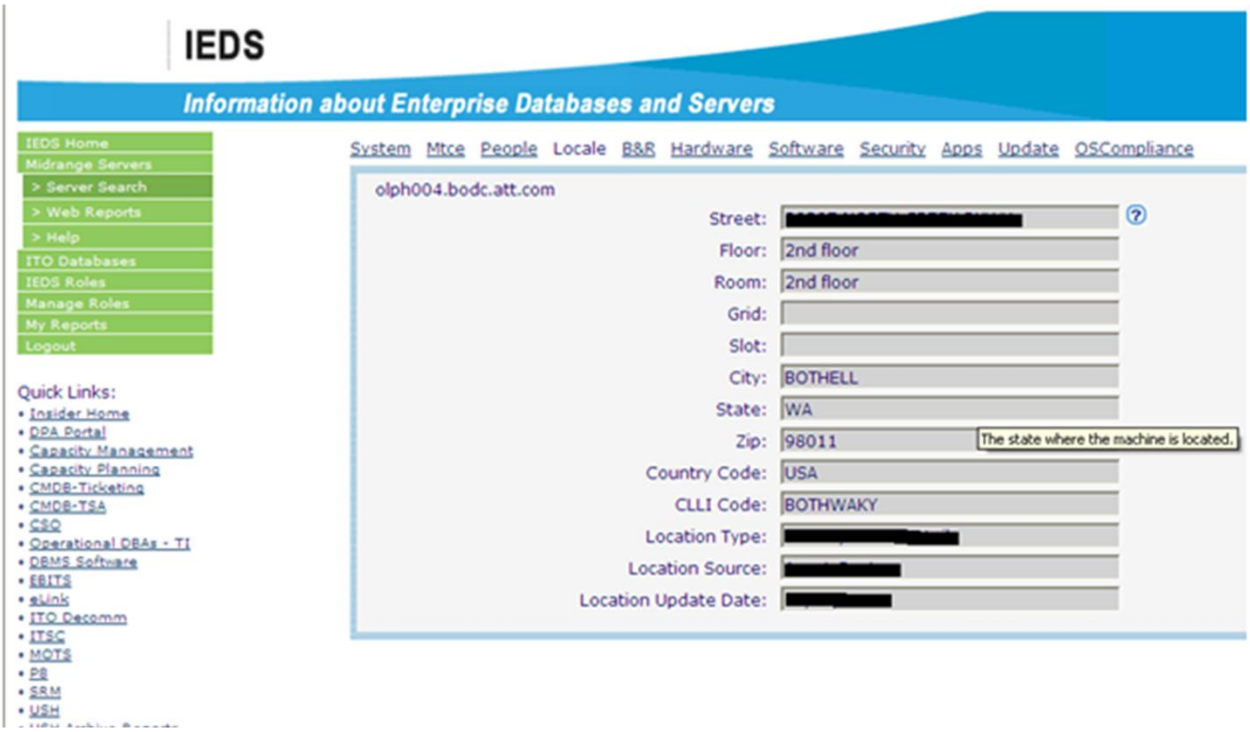
- System name: Same name that we enter.
- Status : Production , Development or Test.
- Alias : Another name.
- Host-ID : Unique ID fo every server.
- Vendor : Maker of the Machine(DELL or HP etc).
- Operating system (Linux,Hp – UX ,AIX)
- Release : Currently installed release level of the operating system.
- In Service : Y/N .
- ITO managed : Y/N .
- Global status : Active or Expired.
- Domain : domain name .

- System type : Standalone or clustered.
- Amity : Y/N .
- Patrol : Y/N .

6.1.2 Locale

The image below shows the interface when we click on locale tab.

This information is generally required when the problem in the hardware related. The employee has to contact the engineer and the address as well as the information regarding the cause has to be conveyed. This needs consultation with onshore system administrator and team lead and their contact information is retrieved by clicking on the people tab.



olph004.bodc.att.com

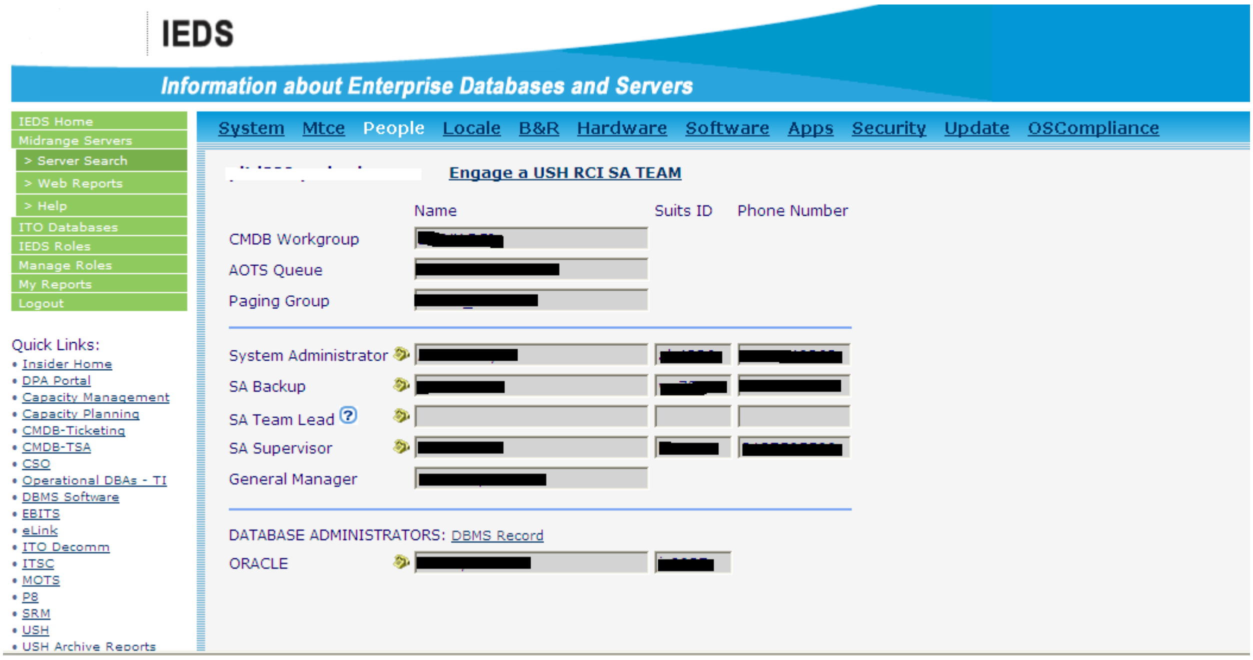
Street:	[redacted]
Floor:	2nd floor
Room:	2nd floor
Grid:	
Slot:	
City:	BOTHELL
State:	WA
Zip:	98011
Country Code:	USA
CLLI Code:	BOTHWAKY
Location Type:	[redacted]
Location Source:	[redacted]
Location Update Date:	[redacted]

6.1.3 People

The next image is that which contains all the information regarding people who are somehow related to this particular server.

The name of the people along with their AT&T ID and contact number is displayed. As I mentioned before it is must for any offshore employee to check the name of the SA Team Lead before logging into any server (as per client norms)

If the name is not of one’s SA Team Lead then the ticket has to be reverted back to EMOC team along with information.



[redacted] Engage a USH RCI SA TEAM

	Name	Suits ID	Phone Number
CMDB Workgroup	[redacted]		
AOTS Queue	[redacted]		
Paging Group	[redacted]		
System Administrator	[redacted]	[redacted]	[redacted]
SA Backup	[redacted]	[redacted]	[redacted]
SA Team Lead ?			
SA Supervisor	[redacted]	[redacted]	[redacted]
General Manager	[redacted]		

DATABASE ADMINISTRATORS: DBMS Record

ORACLE	[redacted]	[redacted]

6.1.4 Hardware

It has following information :

- Total memory: The total amount of physical memory installed.
- CPU Architecture
- 32 /64 Bit: Capability of CPU in Bits.
- CPU speed
- CPU’s: No of CPU’s installed on the system.

6.1.5 Software

Lists all the application packages installed on the system along with the time of installation and the name and ID of the person who has installed the package.

6.1.5 Security

Contains information about the security policies used and installed in the system.

6.1.6 Updates

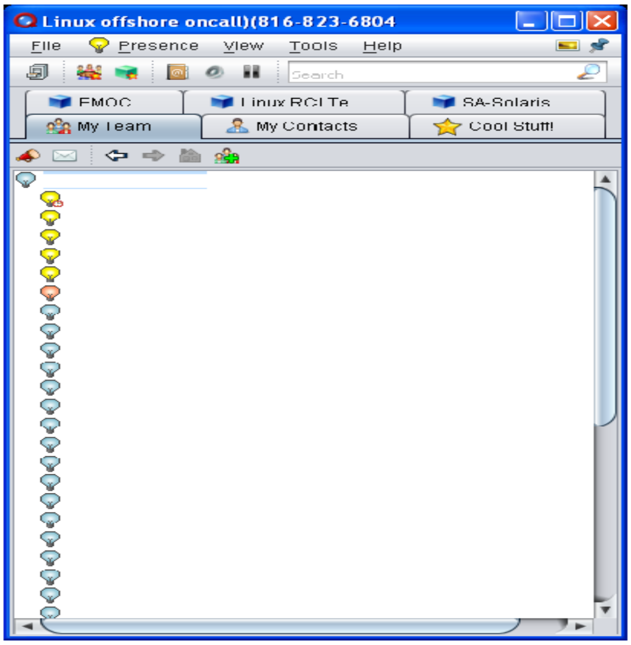
All latest updates that have taken place in the system

Last system update date / ID

6.2 Q Team-Link Messenger

Q Team Link Messenger is messaging tool where we can create our own group with our appropriate group members to communicate with each other through messages. This is a highly supportive tool (highly useful for beginners) as solving a ticket may require help from a team member or even some help from members of other teams.

It is instant messaging tool, we can ping our concern team member at that time when he is available on Q Messenger .So in a way it prevents a lot of commotion which is sure to take place if such a tool is not available.



As we can see in the snapshot above one can along with his own team can contact with team of LINUX RCI , SA – Solaris, EMOC team etc.

6.3 Jump Point

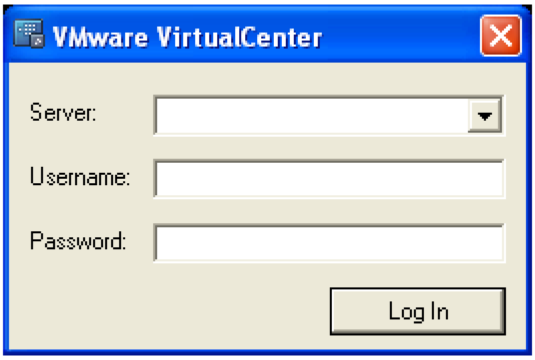
Jump Point is nothing but a server that is used to login to other servers without actually issuing a password. This is meant for convenience and security purpose as all do not have the root password and only the users having access to Jump Point can login to other boxes.

6.4 VMware Virtual Center

1) You will get a vsphere client installed onto your desktop

2) Double click the installer package and when you get to the point where you are prompted for what to install, select the Virtual Center Client **ONLY**

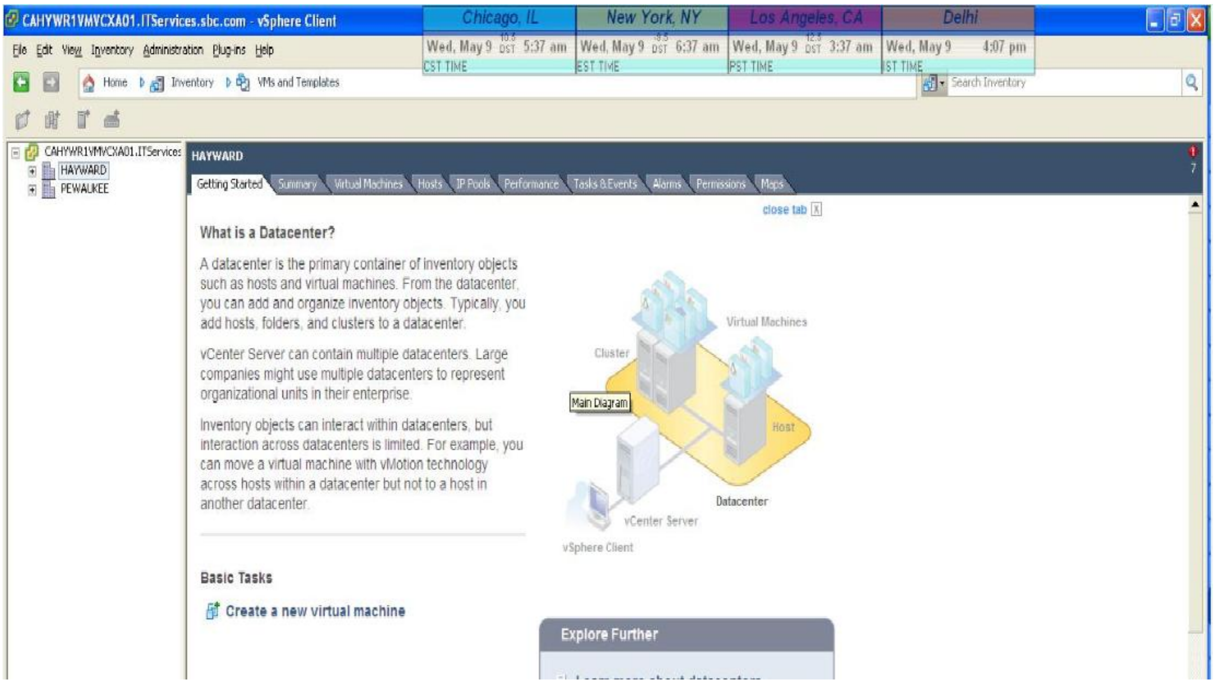
3) Once the client is installed launch VM Ware Virtual Center. A box like this one will appear:



4) There are two options for the servers.
   The test/TIL server is: “WIWAUK4VMVCXA01” (only use this if you are told to do so)
   The production Virtual Center server is: “CASNDG1VMVCXA0”

5) For username, you will need to enter your domain and username that you use on your desktop or laptop:
   ie: ITSERVICES\dk411w

6) The password will be the same as you use to login to your desktop or laptop as well.
   NOTE: If you cannot login or if you do not have the permissions that you believe you should have, send an email to vmware@att.com



CAHYWR1VMVCXA01.ITServices.sbc.com - vSphere Client
Chicago, IL
Wed, May 9 5:37 am CST TIME
New York, NY
Wed, May 9 6:37 am EST TIME
Los Angeles, CA
Wed, May 9 3:37 am PST TIME
Delhi
Wed, May 9 4:07 pm IST TIME
File Edit View Inventory Administration Plug-ins Help
Home Inventory VMs and Templates
Search Inventory
CAHYWR1VMVCXA01.ITServices
HAYWARD
PEWAUKEE
HAYWARD
Getting Started Summary Virtual Machines Hosts IP Pools Performance Tasks & Events Alarms Permissions Maps
close tab
Virtual Machines
Cluster
Main Diagram
Host
Datacenter
vCenter Server
vSphere Client

What is a Datacenter?

A datacenter is the primary container of inventory objects such as hosts and virtual machines. From the datacenter, you can add and organize inventory objects. Typically, you add hosts, folders, and clusters to a datacenter.

vCenter Server can contain multiple datacenters. Large companies might use multiple datacenters to represent organizational units in their enterprise.

Inventory objects can interact within datacenters, but interaction across datacenters is limited. For example, you can move a virtual machine with vMotion technology across hosts within a datacenter but not to a host in another datacenter.

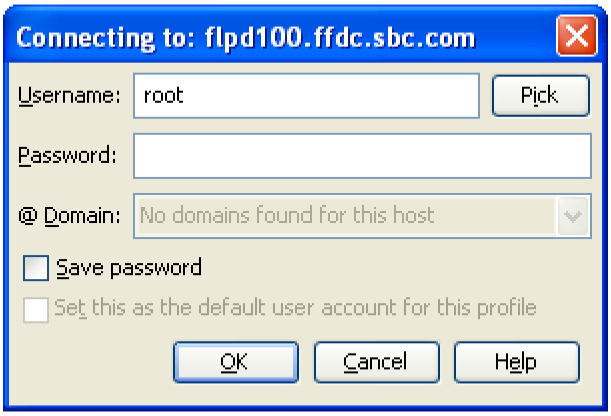
Basic Tasks

Create a new virtual machine

Explore Further

6.5 VXVM GUI:

This tool is used to manage the Veritas Mounted File Systems OR Veritas Volume Manager (VxVM). The GUI interface is as shown below:



Connecting to: flpd100.ffdc.sbc.com

Username: root

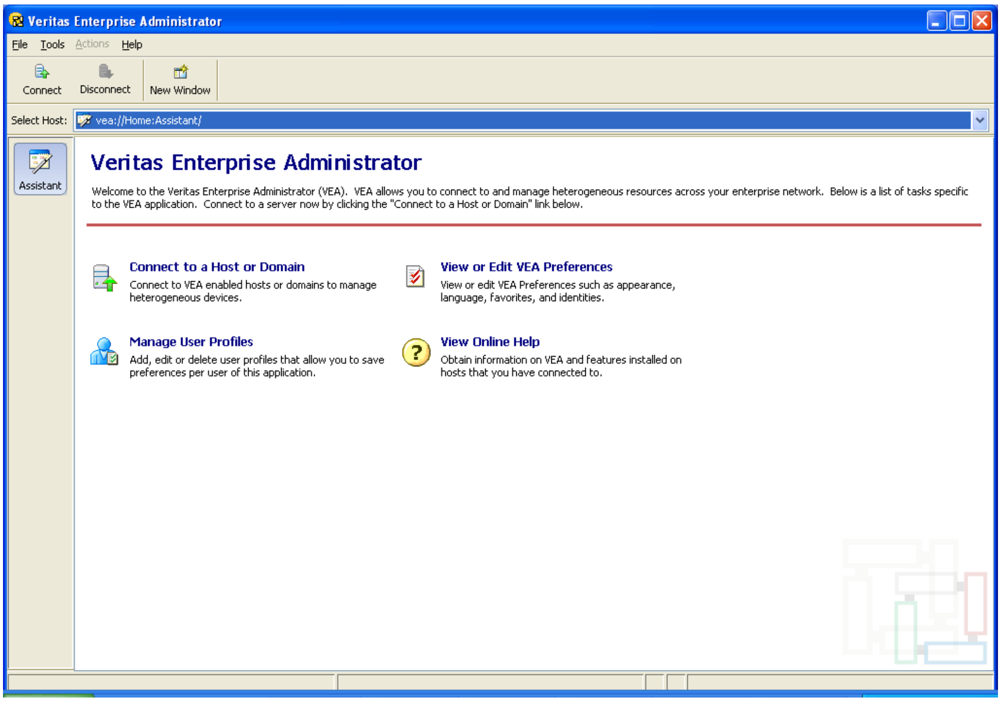
Password:

@ Domain: No domains found for this host

☐ Save password

☐ Set this as the default user account for this profile

Enter the root login credentials to connect to the VEA GUI. You can then manage the FS resizing/modifying from there.



Select Host: vea://Home:Assistant/

Veritas Enterprise Administrator

Welcome to the Veritas Enterprise Administrator (VEA). VEA allows you to connect to and manage heterogeneous resources across your enterprise network. Below is a list of tasks specific to the VEA application. Connect to a server now by clicking the "Connect to a Host or Domain" link below.

Connect to a Host or Domain
Connect to VEA enabled hosts or domains to manage heterogeneous devices.

View or Edit VEA Preferences
View or edit VEA Preferences such as appearance, language, favorites, and identities.

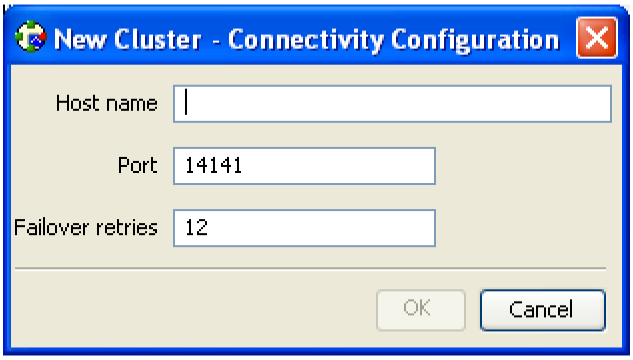
Manage User Profiles
Add, edit or delete user profiles that allow you to save preferences per user of this application.

View Online Help
Obtain information on VEA and features installed on hosts that you have connected to.

Select the **Connect to a Host or Domain** option:

6.6 VCS GUI:

This GUI too is used to manage servers in Veritas Cluster. Click on the New Cluster option in the File dropdown menu to connect to a cluster:



Host name

Port 14141

Failover retries 12

Enter the hostname and then the credentials to connect to the cluster. Once you are logged in you will be able to switch over the groups, failover them, etc.

6.7 BMC Remedy User (AOTS)

Remedy User is a client software application developed by BMC software that allows a user to interact with Action Request system based applications. With this software a user can submit, modify, and search records within Action Request System. It is a Windows based software and must be installed locally on the user's desktop. Remedy

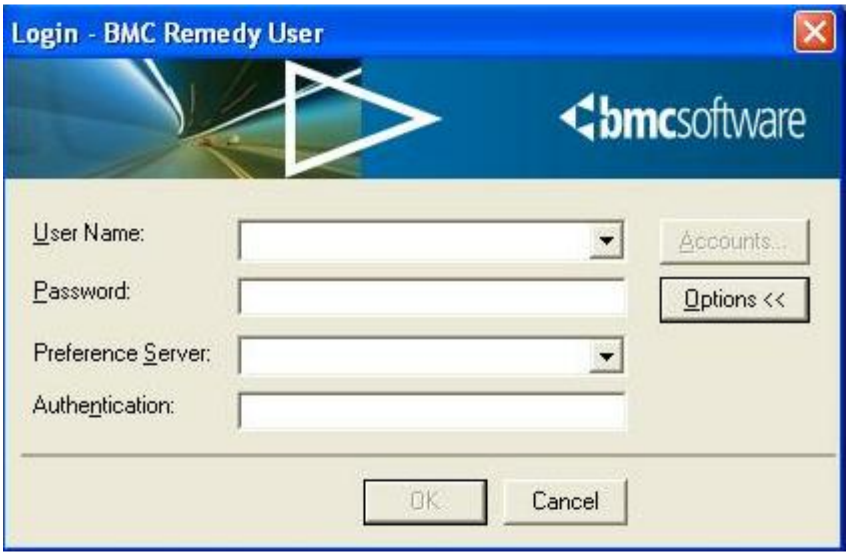
Reduce complexity and make customer support, change, asset, and request management a seamless integrated process.

This comprehensive suite includes:

A full set of IT service management applications that share a native, purpose-built architecture and best-practice process flows.

- The industry's leading service desk solution
- A closed-loop change and release process tied to incidents and problems
- Self-service request catalog for IT, security, and, business needs
- Tracking of incident response times and service desk performance against SLAs
- Asset and software license lifecycle and compliance management
- Real-time performance and ROI metrics reporting
- Mobile applications with instant alerts

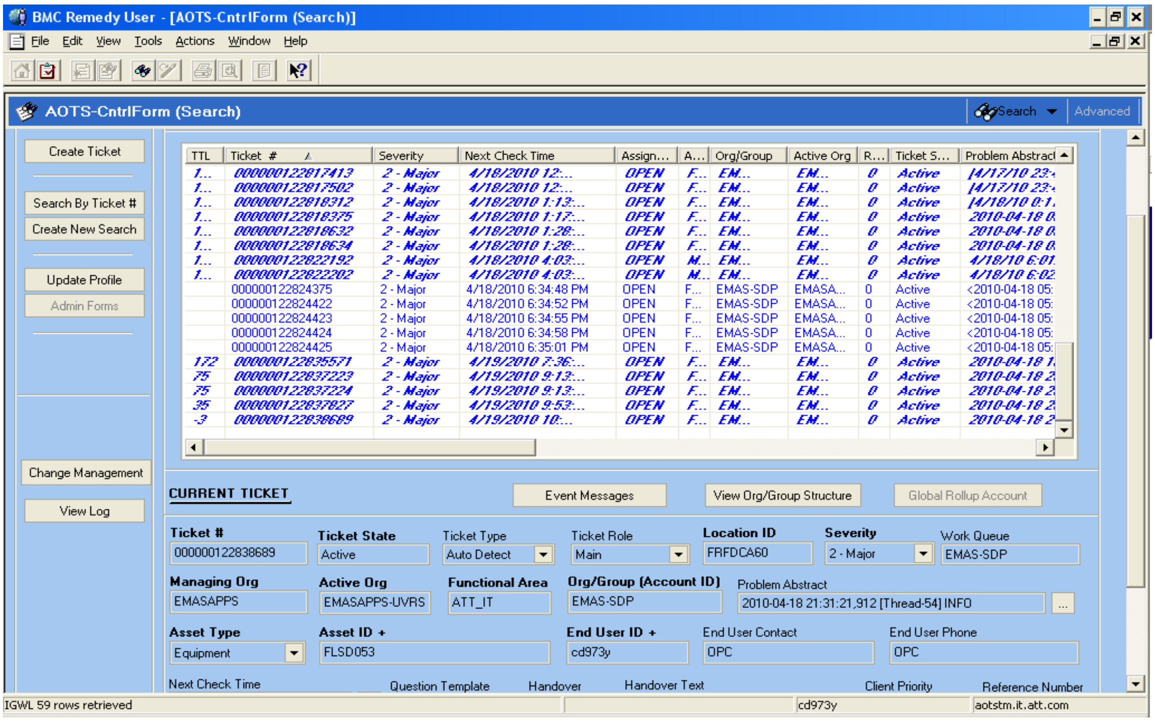
Also Know As AOTS. It Is An Online Transaction/Ticketing System Provided By AT&T Which Is Used To Monitor, Create Or Close The Tickets. Whenever There Is An Exception, Failure Or Error, A New Ticket Is Created With Necessary Details Required To Troubleshoot The Issue.



This is login window. Here you enter the username and password along with the preference server.

Whenever there is an alert generated or the end user faces a problem a ticket is created on this tool. Ticket is generated by the EMOC team ,sometimes the ticket is generated by the end user itself in this case the ticket is called **Internal Engaguement.**

After the login the window like the one on the next page is opened. Here you can see the list of all the tickets that are present on the remedy at present.



TTL	Ticket #	Severity	Next Check Time	Assign...	A...	Org/Group	Active Org	R...	Ticket S...	Problem Abstract
1...	000000122817413	2 - Major	4/18/2010 12:...	OPEN	F...	EM...	EM...	0	Active	[4/17/10 23:4
1...	000000122817502	2 - Major	4/18/2010 12:...	OPEN	F...	EM...	EM...	0	Active	[4/17/10 23:4
1...	000000122818312	2 - Major	4/18/2010 1:13:...	OPEN	F...	EM...	EM...	0	Active	[4/18/10 0:1.
1...	000000122818375	2 - Major	4/18/2010 1:17:...	OPEN	F...	EM...	EM...	0	Active	2010-04-18 0.
1...	000000122818632	2 - Major	4/18/2010 1:28:...	OPEN	F...	EM...	EM...	0	Active	2010-04-18 0.
1...	000000122818634	2 - Major	4/18/2010 1:28:...	OPEN	F...	EM...	EM...	0	Active	2010-04-18 0.
1...	000000122822192	2 - Major	4/18/2010 4:03:...	OPEN	M...	EM...	EM...	0	Active	4/18/10 6:01.
1...	000000122822202	2 - Major	4/18/2010 4:03:...	OPEN	M...	EM...	EM...	0	Active	4/18/10 6:02
	000000122824375	2 - Major	4/18/2010 6:34:48 PM	OPEN	F...	EMAS-SDP	EMASA...	0	Active	<2010-04-18 05:
	000000122824422	2 - Major	4/18/2010 6:34:52 PM	OPEN	F...	EMAS-SDP	EMASA...	0	Active	<2010-04-18 05:
	000000122824423	2 - Major	4/18/2010 6:34:55 PM	OPEN	F...	EMAS-SDP	EMASA...	0	Active	<2010-04-18 05:
	000000122824424	2 - Major	4/18/2010 6:34:58 PM	OPEN	F...	EMAS-SDP	EMASA...	0	Active	<2010-04-18 05:
	000000122824425	2 - Major	4/18/2010 6:35:01 PM	OPEN	F...	EMAS-SDP	EMASA...	0	Active	<2010-04-18 05:
172	000000122835571	2 - Major	4/19/2010 7:36:...	OPEN	F...	EM...	EM...	0	Active	2010-04-18 1.
75	000000122837223	2 - Major	4/19/2010 9:13:...	OPEN	F...	EM...	EM...	0	Active	2010-04-18 2.
75	000000122837224	2 - Major	4/19/2010 9:13:...	OPEN	F...	EM...	EM...	0	Active	2010-04-18 2.
35	000000122837827	2 - Major	4/19/2010 9:53:...	OPEN	F...	EM...	EM...	0	Active	2010-04-18 2.
-3	000000122838689	2 - Major	4/19/2010 10:...	OPEN	F...	EM...	EM...	0	Active	2010-04-18 2

CURRENT TICKET

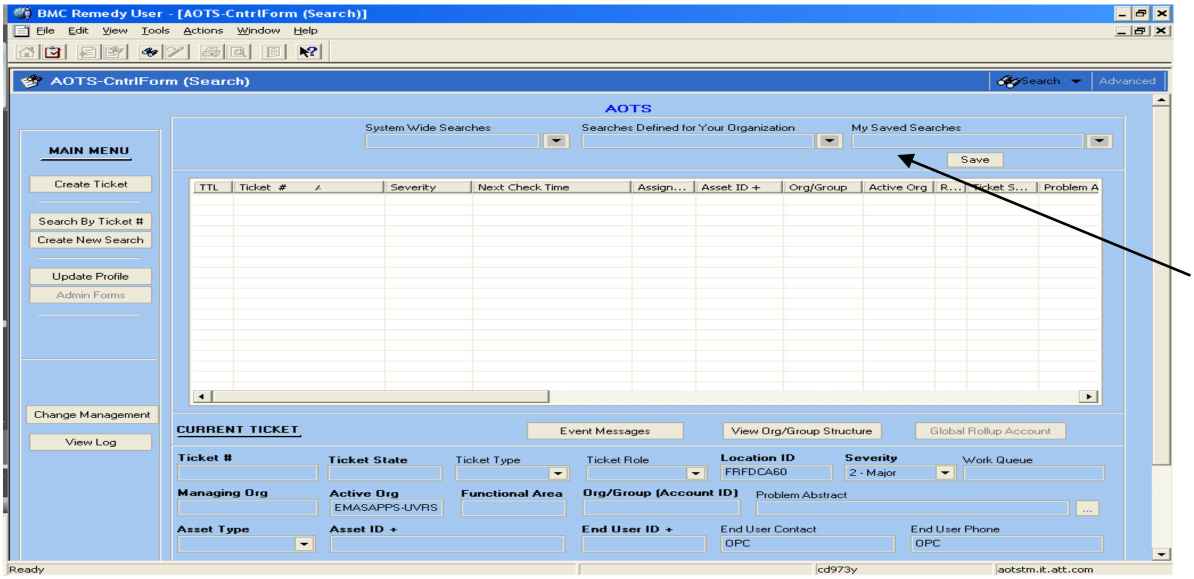
Ticket # 000000122838689
Ticket State Active
Ticket Type Auto Detect
Ticket Role Main
Location ID FRFDCA60
Severity 2 - Major
Work Queue EMAS-SDP
Managing Org EMASAPPS
Active Org EMASAPPS-UVRS
Functional Area ATT_IT
Org/Group (Account ID) EMAS-SDP
Problem Abstract 2010-04-18 21:31:21,912 [Thread-54] INFO
Asset Type Equipment
Asset ID + FLSD053
End User ID + cd973y
End User Contact OPC
End User Phone OPC
Next Check Time
Question Template
Handover
Handover Text
Client Priority
Reference Number

IGWL 59 rows retrieved | cd973y | aotstm.it.att.com

6.7.1 Creating profile

First thing when you use remedy user for first time is to create a profile. Steps to create a new profile:Click on “search by ticket” option.

A new window opens where one enters his/her Active org (e.g.USH –LINUX) and group(USH).



Location ID: FRFDCA60

Severity: 2 - Major

Active Org: EMASAPPS-UVRS

End User Contact: OPC

End User Phone: OPC

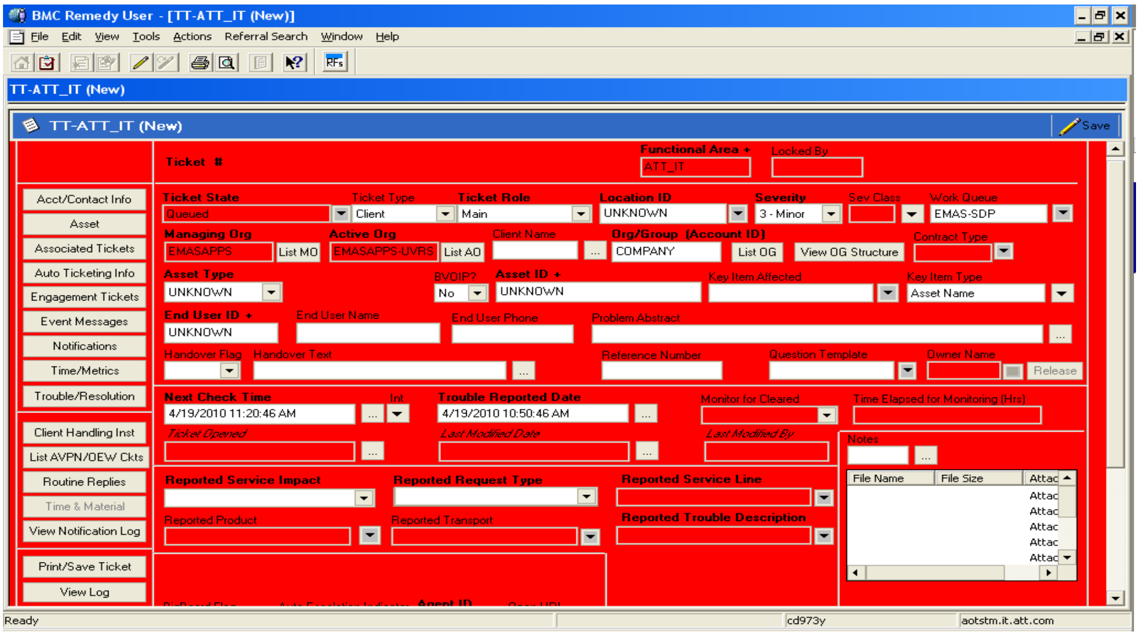
Ready | cd973y | aotstm.it.att.com

6.7.2 How to Create an AOTS ticket

After creating the profile click on “My saved searches” list option. All the tickets with different groups are present here. After that one can create his own saved search.

Steps to create AOTS ticket:

1. Click on the “Create new ticket” option below “Main menu”.



Functional Area +: ATT_IT

Ticket State: Queued

Ticket Type: Client

Ticket Role: Main

Location ID: UNKNOWN

Severity: 3 - Minor

Work Queue: EMAS-SDP

Managing Org: EMASAPPS

Active Org: EMASAPPS-UVRS

Org/Group (Account ID): COMPANY

Asset Type: UNKNOWN

BVOIP?: No

Asset ID +: UNKNOWN

Key Item Type: Asset Name

End User ID +: UNKNOWN

Next Check Time: 4/19/2010 11:20:46 AM

Trouble Reported Date: 4/19/2010 10:50:46 AM

Ready | cd973y | aotstm.it.att.com

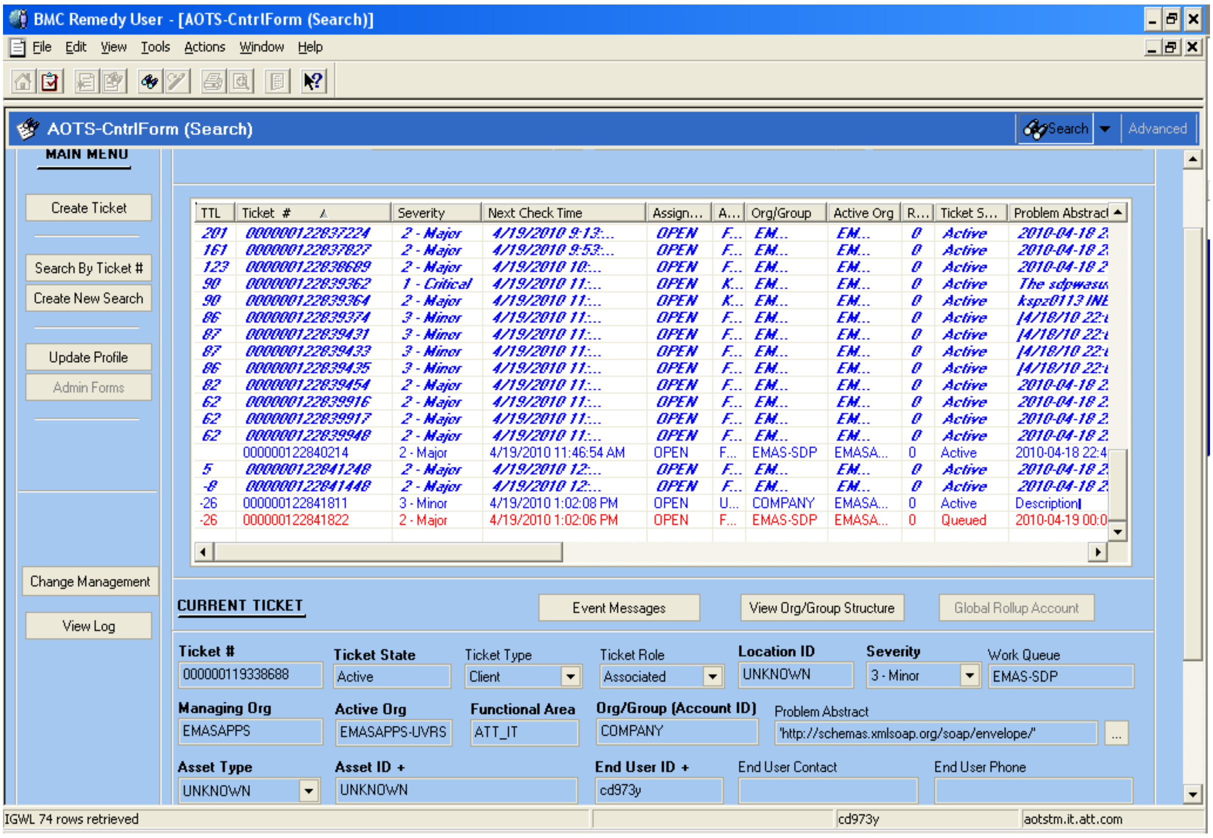
Ticket information page.

This window is where we enter the details of the ticket which is to be created

- Ticket state –Active, Ready To Close, Closed, Deferred, Queued.

- Key Item affected – Server Name.
- End User ID – AT&T ID of the user affected.
- Problem Abstract – Some information about the problem.
- Severity – 1-critical, 2-major, 3-minor.
- Reported service impact – No or Yes.
- Reported Service line – Application , operating system ,Storage etc.
- Reported trouble description- Alarm.

2. Click on the save button and the ticket’s created.
3. After this a ticket number is created and provided to you.
4. Close the ticket from a button at right bottom of the screen
5. Then when you click the “search” button on main page, the ticket that you have created will be shown.



TTL	Ticket #	Severity	Next Check Time	Assign...	A...	Org/Group	Active Org	R...	Ticket S...	Problem Abstract
201	000000122837224	2 - Major	4/19/2010 9:13:...	OPEN	F...	EM...	EM...	0	Active	2010-04-18 2
161	000000122837827	2 - Major	4/19/2010 9:53:...	OPEN	F...	EM...	EM...	0	Active	2010-04-18 2
123	000000122838689	2 - Major	4/19/2010 10:...	OPEN	F...	EM...	EM...	0	Active	2010-04-18 2
90	000000122839362	1 - Critical	4/19/2010 11:...	OPEN	K...	EM...	EM...	0	Active	The sdpwasu
90	000000122839364	2 - Major	4/19/2010 11:...	OPEN	K...	EM...	EM...	0	Active	kspz0113 INE
86	000000122839374	3 - Minor	4/19/2010 11:...	OPEN	F...	EM...	EM...	0	Active	|4/18/10 22:
87	000000122839431	3 - Minor	4/19/2010 11:...	OPEN	F...	EM...	EM...	0	Active	|4/18/10 22:
87	000000122839433	3 - Minor	4/19/2010 11:...	OPEN	F...	EM...	EM...	0	Active	|4/18/10 22:
86	000000122839435	3 - Minor	4/19/2010 11:...	OPEN	F...	EM...	EM...	0	Active	|4/18/10 22:
82	000000122839454	2 - Major	4/19/2010 11:...	OPEN	F...	EM...	EM...	0	Active	2010-04-18 2
62	000000122839916	2 - Major	4/19/2010 11:...	OPEN	F...	EM...	EM...	0	Active	2010-04-18 2
62	000000122839917	2 - Major	4/19/2010 11:...	OPEN	F...	EM...	EM...	0	Active	2010-04-18 2
62	000000122839948	2 - Major	4/19/2010 11:...	OPEN	F...	EM...	EM...	0	Active	2010-04-18 2
	000000122840214	2 - Major	4/19/2010 11:46:54 AM	OPEN	F...	EMAS-SDP	EMASA...	0	Active	2010-04-18 22:4
5	000000122841248	2 - Major	4/19/2010 12:...	OPEN	F...	EM...	EM...	0	Active	2010-04-18 2
-8	000000122841448	2 - Major	4/19/2010 12:...	OPEN	F...	EM...	EM...	0	Active	2010-04-18 2
-26	000000122841811	3 - Minor	4/19/2010 1:02:08 PM	OPEN	U...	COMPANY	EMASA...	0	Active	Description
-26	000000122841822	2 - Major	4/19/2010 1:02:06 PM	OPEN	F...	EMAS-SDP	EMASA...	0	Queued	2010-04-19 00:0

Ticket # 000000119338688
Ticket State Active
Ticket Type Client
Ticket Role Associated
Location ID UNKNOWN
Severity 3 - Minor
Work Queue EMAS-SDP
Managing Org EMASAPPS
Active Org EMASAPPS-UVRS
Functional Area ATT_IT
Org/Group (Account ID) COMPANY
Problem Abstract 'http://schemas.xmlsoap.org/soap/envelope/'
Asset Type UNKNOWN
Asset ID + UNKNOWN
End User ID + cd973y
End User Contact
End User Phone

IGWL 74 rows retrieved cd973y aotstm.it.att.com

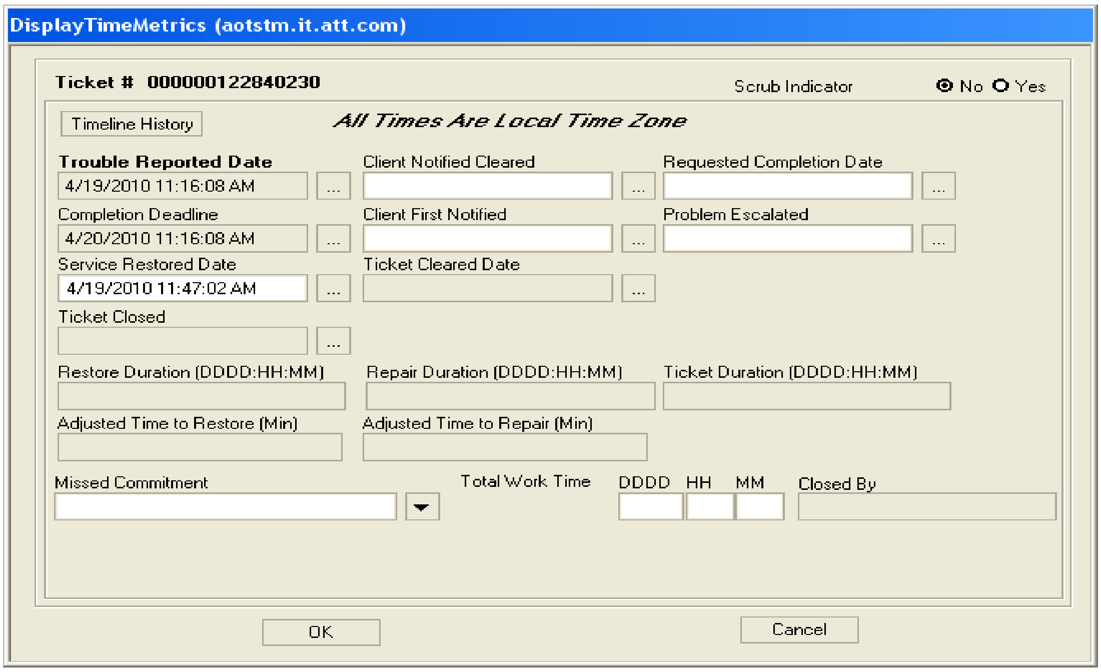
6.7.3 How to close AOTS ticket

There are six things that need to be done while closing an AOTS ticket.

1. Click on the “time metrics” option that is present on the “ticket information page”.

- A new window opens where the time has to be filled.
- The trouble reported time and completion deadline are already there and the system administrator who has accepted the ticket has to fill the “service restored time”.

(completion deadline is mentioned accoding to the SLA agreements)



DisplayTimeMetrics (aotstm.it.att.com)

Ticket # 000000122840230

Scrub Indicator ● No ○ Yes

All Times Are Local Time Zone

Trouble Reported Date 4/19/2010 11:16:08 AM

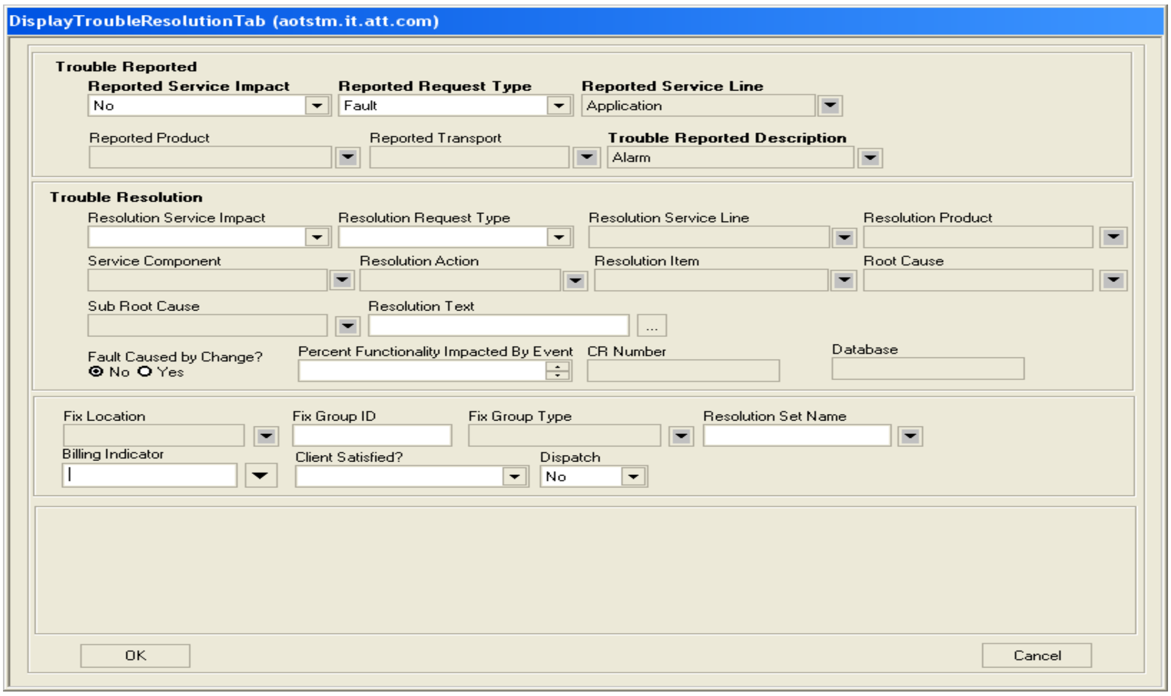
Completion Deadline 4/20/2010 11:16:08 AM

Service Restored Date 4/19/2010 11:47:02 AM

Timeline History
Client Notified Cleared
Requested Completion Date
Client First Notified
Problem Escalated
Ticket Cleared Date
Ticket Closed
Restore Duration (DDDD:HH:MM)
Repair Duration (DDDD:HH:MM)
Ticket Duration (DDDD:HH:MM)
Adjusted Time to Restore (Min)
Adjusted Time to Repair (Min)
Missed Commitment
Total Work Time
DDDD
HH
MM
Closed By
OK
Cancel

2. Click on the “trouble resolution” on ticket information page .

- Resolution Text = Mention The Steps Taken To Address The Alert (OA = Only Require If There Is An Outage)
- Resolution Set Name = “Application – Known Issue”
- Fix Group Type = “EMAS Only”
- Fix Group ID =“EMAS SDP”
- Fix Location = “Other”



DisplayTroubleResolutionTab (aotstm.it.att.com)

Trouble Reported

Reported Service Impact: No
Reported Request Type: Fault
Reported Service Line: Application
Reported Product:
Reported Transport:
Trouble Reported Description: Alarm

Trouble Resolution

Resolution Service Impact:
Resolution Request Type:
Resolution Service Line:
Resolution Product:
Service Component:
Resolution Action:
Resolution Item:
Root Cause:
Sub Root Cause:
Resolution Text:
Fault Caused by Change? ● No ○ Yes
Percent Functionality Impacted By Event:
CR Number:
Database:

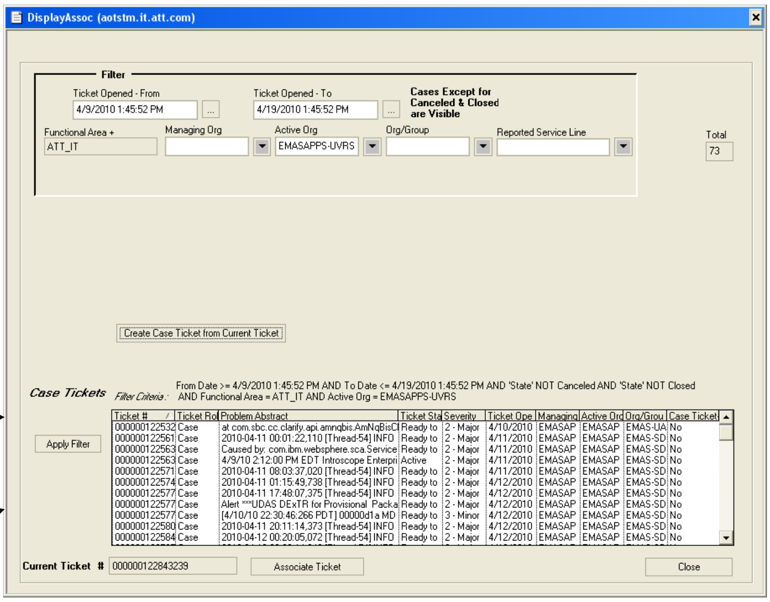
Fix Location:
Fix Group ID:
Fix Group Type:
Resolution Set Name:
Billing Indicator:
Client Satisfied?
Dispatch: No

3. Close the ticket by clicking on the “ticket state”.
4. Finally save the ticket and close.

6.7.4 How to Associate AOTS tickets.

Association of the tickets needs to be done in the case when different users have created tickets on the same issue.

1. Double click a ticket to open it.
2. Click on the “Associated tickets” option on the Ticket information page. A new window like this opens.



DisplayAssoc (aotstm.it.att.com)

Filter

Ticket Opened - From: 4/9/2010 1:45:52 PM
Ticket Opened - To: 4/19/2010 1:45:52 PM
Cases Except for Canceled & Closed are Visible
Functional Area +: ATT_IT
Managing Org:
Active Org: EMASAPPS-UVRS
Org/Group:
Reported Service Line:
Total: 73

Case Tickets

Filter Criteria : From Date >= 4/9/2010 1:45:52 PM AND To Date <= 4/19/2010 1:45:52 PM AND 'State' NOT Canceled AND 'State' NOT Closed AND Functional Area = ATT_IT AND Active Org = EMASAPPS-UVRS

Ticket #	Ticket Rol	Problem Abstract	Ticket Sta	Severity	Ticket Ope	Managing	Active Org	Org/Grou	Case Ticket
000000122532	Case	at com.sbc.cc.clarify.api.amnqbis.AmNqBisCl	Ready to	2 - Major	4/10/2010	EMASAP	EMASAP	EMAS-UA	No
000000122561	Case	2010-04-11 00:01:22,110 [Thread-54] INFO	Ready to	2 - Major	4/11/2010	EMASAP	EMASAP	EMAS-SD	No
000000122563	Case	Caused by: com.ibm.websphere.sca.Service	Ready to	2 - Major	4/11/2010	EMASAP	EMASAP	EMAS-SD	No
000000122563	Case	4/9/10 2:12:00 PM EDT Introscope Enterpri	Active	2 - Major	4/11/2010	EMASAP	EMASAP	EMAS-SD	No
000000122571	Case	2010-04-11 08:03:37,020 [Thread-54] INFO	Ready to	2 - Major	4/11/2010	EMASAP	EMASAP	EMAS-SD	No
000000122574	Case	2010-04-11 01:15:49,738 [Thread-54] INFO	Ready to	2 - Major	4/12/2010	EMASAP	EMASAP	EMAS-SD	No
000000122577	Case	2010-04-11 17:48:07,375 [Thread-54] INFO	Ready to	2 - Major	4/12/2010	EMASAP	EMASAP	EMAS-SD	No
000000122577	Case	Alert ***UDAS DExTR for Provisional Packa	Ready to	2 - Major	4/12/2010	EMASAP	EMASAP	EMAS-SD	No
000000122577	Case	[4/10/10 22:30:46:266 PDT] 00000d1a MD	Ready to	3 - Minor	4/12/2010	EMASAP	EMASAP	EMAS-SD	No
000000122580	Case	2010-04-11 20:11:14,373 [Thread-54] INFO	Ready to	2 - Major	4/12/2010	EMASAP	EMASAP	EMAS-SD	No
000000122584	Case	2010-04-12 00:20:05,072 [Thread-54] INFO	Ready to	2 - Major	4/12/2010	EMASAP	EMASAP	EMAS-SD	No

Current Ticket # 000000122843239

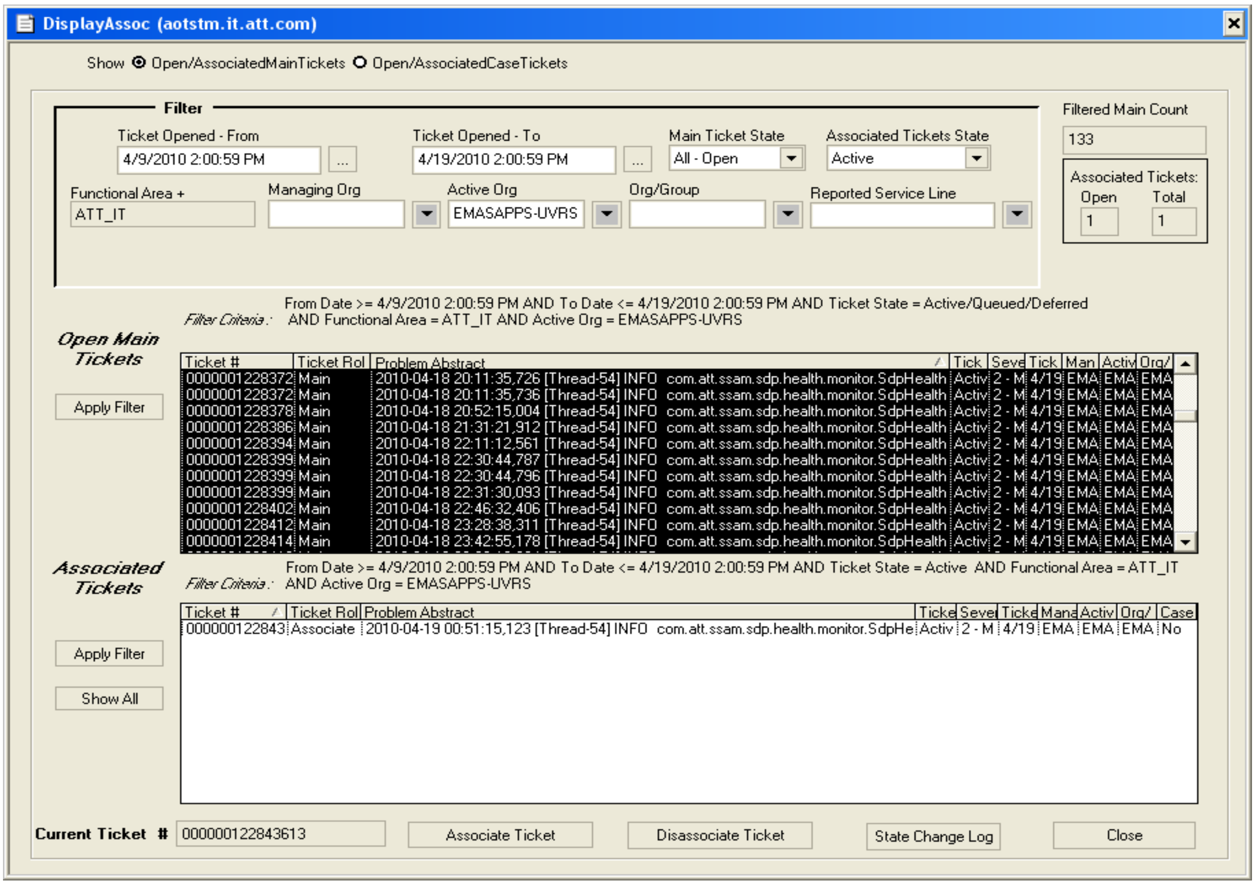
Create case ticket from current ticket

Filter criteria

3. Click on the option “create case ticket from current ticket” and a new ticket is created. You will notice that its “ticket role” changes from ‘main’ to ‘case’.
4. Now again click on the “Associated tickets” option and you will see that a different window opens. Then you select tickets from main tickets to associate them under the ticket that you have just created.

Filter criteria : Filter criteria is used to filter the main tickets.

For e.g.- From Date>=02/03/2012 2:00:22PM AND To Date<=02/03/2012 3:23:12PM AND Ticket State = Active/Queued/Deferred AND Functional Area=AT&T IT AND Active Org=USH-LINUX.



Show ● Open/AssociatedMainTickets ○ Open/AssociatedCaseTickets

Ticket Opened - From: 4/9/2010 2:00:59 PM

Ticket Opened - To: 4/19/2010 2:00:59 PM

Main Ticket State: All - Open

Associated Tickets State: Active

Filtered Main Count: 133

Functional Area +: ATT_IT

Managing Org:

Active Org: EMASAPPS-UVRS

Org/Group:

Reported Service Line:

Associated Tickets: Open 1 Total 1

Filter Criteria : From Date >= 4/9/2010 2:00:59 PM AND To Date <= 4/19/2010 2:00:59 PM AND Ticket State = Active/Queued/Deferred AND Functional Area = ATT_IT AND Active Org = EMASAPPS-UVRS

Ticket #	Ticket Rol	Problem Abstract	Tick	Seve	Tick	Man	Activ	Org/
0000001228372	Main	2010-04-18 20:11:35,726 [Thread-54] INFO com.att.ssam.sdp.health.monitor.SdpHealth	Activ	2 - M	4/19	EMA	EMA	EMA
0000001228372	Main	2010-04-18 20:11:35,736 [Thread-54] INFO com.att.ssam.sdp.health.monitor.SdpHealth	Activ	2 - M	4/19	EMA	EMA	EMA
0000001228378	Main	2010-04-18 20:52:15,004 [Thread-54] INFO com.att.ssam.sdp.health.monitor.SdpHealth	Activ	2 - M	4/19	EMA	EMA	EMA
0000001228386	Main	2010-04-18 21:31:21,912 [Thread-54] INFO com.att.ssam.sdp.health.monitor.SdpHealth	Activ	2 - M	4/19	EMA	EMA	EMA
0000001228394	Main	2010-04-18 22:11:12,561 [Thread-54] INFO com.att.ssam.sdp.health.monitor.SdpHealth	Activ	2 - M	4/19	EMA	EMA	EMA
0000001228399	Main	2010-04-18 22:30:44,787 [Thread-54] INFO com.att.ssam.sdp.health.monitor.SdpHealth	Activ	2 - M	4/19	EMA	EMA	EMA
0000001228399	Main	2010-04-18 22:30:44,796 [Thread-54] INFO com.att.ssam.sdp.health.monitor.SdpHealth	Activ	2 - M	4/19	EMA	EMA	EMA
0000001228399	Main	2010-04-18 22:31:30,093 [Thread-54] INFO com.att.ssam.sdp.health.monitor.SdpHealth	Activ	2 - M	4/19	EMA	EMA	EMA
0000001228402	Main	2010-04-18 22:46:32,406 [Thread-54] INFO com.att.ssam.sdp.health.monitor.SdpHealth	Activ	2 - M	4/19	EMA	EMA	EMA
0000001228412	Main	2010-04-18 23:28:38,311 [Thread-54] INFO com.att.ssam.sdp.health.monitor.SdpHealth	Activ	2 - M	4/19	EMA	EMA	EMA
0000001228414	Main	2010-04-18 23:42:55,178 [Thread-54] INFO com.att.ssam.sdp.health.monitor.SdpHealth	Activ	2 - M	4/19	EMA	EMA	EMA

Filter Criteria : From Date >= 4/9/2010 2:00:59 PM AND To Date <= 4/19/2010 2:00:59 PM AND Ticket State = Active AND Functional Area = ATT_IT AND Active Org = EMASAPPS-UVRS

Ticket #	Ticket Rol	Problem Abstract	Ticke	Seve	Ticke	Mana	Activ	Org/	Case
000000122843	Associate	2010-04-19 00:51:15,123 [Thread-54] INFO com.att.ssam.sdp.health.monitor.SdpHe	Activ	2 - M	4/19	EMA	EMA	EMA	No

Current Ticket # 000000122843613

Finally click on close button.

7. CRON

Cron is a daemon that executes scheduled commands. Cron is started automatically from /etc/init.d on entering multi-user runlevels. Cron searches its spool area (/var/spool/cron/crontabs) for crontab files (which are named after accounts in / etc/passwd); crontabs found are loaded into memory. Note that crontabs in this directory should not be accessed directly - the crontab command should be used to access and update them.

Cron also reads /etc/crontab, which is in a slightly different format. Additionally, cron reads the files in /etc/cron.d.

Cron then wakes up every minute, examining all stored crontabs, checking each command to see if it should be run in the current minute. When executing commands, any output is mailed to the owner of the crontab (or to the user named in the MAILTO environment variable in the crontab, if such exists). The children copies of cron running these processes have their name coerced to uppercase, as will be seen in the syslog and ps output.

Special considerations exist when the clock is changed by less than 3 hours, for example at the beginning and end of daylight savings time. If the time has moved forwards, those jobs which would have run in the time that was skipped will be run soon after the change. Conversely, if the time has moved backwards by less than 3 hours, those jobs that fall into the repeated time will not be re-run.

Only jobs that run at a particular time (not specified as @hourly, nor with '*' in the hour or minute specifier) are affected. Jobs which are specified with wild cards are run based on the new time immediately.

Clock changes of more than 3 hours are considered to be corrections to the clock, and the new time is used immediately.

You should use absolute path names for commands like /bin/ls. This is to insure you call the correct command.

7.1 Crontab Format

Commands are executed by cron when the minute, hour, and month of year fields match the current time, and when at least one of the two day fields (day of month, or day of week) match the current time.

A field may be an asterisk (*), which always stands for "first-last".

Ranges of numbers are allowed. Ranges are two numbers separated with a hyphen. The specified range is inclusive. For example, 8-11 for an "hours" entry specifies execution at hours 8, 9, 10 and 11.

Lists are allowed. A list is a set of numbers (or ranges) separated by commas. Examples: "1,2,5,9", "0-4,8-12".

Step values can be used in conjunction with ranges. Following a range with "/" specifies skips of the number's value through the range. For example, "0-23/2" can be used in the hours field to specify command execution every other hour (the alternative in the V7 standard is "0,2,4,6,8,10,12,14,16,18,20,22"). Steps are also permitted after an asterisk, so if you want to say "every two hours", just use "*/2".

Names can also be used for the "month" and "day of week" fields. Use the first three letters of the particular day or month (case doesn't matter). Ranges or lists of names are not allowed.

The "sixth" field (the rest of the line) specifies the command to be run. The entire command portion of the line, up to a newline or % character, will be executed by /bin/sh or by the shell specified in the SHELL variable of the crontab file. Percent-signs (%) in the command, unless escaped with backslash (\), will be changed into newline characters, and all data after the first % will be sent to the command as standard input. There is no way to split a single command line onto multiple lines, like the shell's trailing "\".

Note: The day of a command's execution can be specified by two fields - day of month, and day of week. If both fields are restricted (i.e., aren't *), the command will be run when either field matches the current time. For example, "30 4 1,15 * 5" would cause a command to be run at 4:30 am on the 1st and 15th of each month, plus every Friday.

Instead of the first five fields, one of eight special strings may appear:

string	meaning
------	-------
@reboot	Run once, at startup.
@yearly	Run once a year, "0 0 1 1 *".

@annually	(same as @yearly)
@monthly	Run once a month, "0 0 1 * *".
@weekly	Run once a week, "0 0 * * 0".
@daily	Run once a day, "0 0 * * *".
@midnight	(same as @daily)
@hourly	Run once an hour, "0 * * * *".

An example of crontab format with commented fields is as follows:

```
# Minute   Hour   Day of Month      Month          Day of Week      Command
# (0-59)   (0-23)     (1-31)    (1-12 or Jan-Dec)  (0-6 or Sun-Sat)
   0          2          12               *                0, 6           /usr/bin/find
```

This line executes the "find" command at 2AM on the 12th of every month that a Sunday or Saturday falls on.

7.2 Examples

Here are some more examples of crontab lines. Use the command "crontab -e" to edit your crontab file.

This line executes the "ping" command every minute of every hour of every day of every month. The standard output is redirected to dev null so we will get no e-mail but will allow the standard error to be sent as a e-mail. If you want no e-mail ever change the command line to "/sbin/ping -c 1 192.168.0.1 > /dev/null 2>&1".

```
*     *     *     *     *     /sbin/ping -c 1 192.168.0.1 > /dev/null
```

This line executes the "ping" and the "ls" command every 12am and 12pm on the 1st day of every 2nd month. It also puts the output of the commands into the log file /var/log/cronrun.

```
0 0,12 1 */2 * /sbin/ping -c 192.168.0.1; ls -la >>/var/log/cronrun
```

This line executes the disk usage command to get the directory sizes every 2am on the 1st through the 10th of each month. E-mail is sent to the email addresses specified with the MAILTO line. The PATH is also set to something different.

```
PATH=/usr/local/sbin:/usr/local/bin:/home/user1/bin
MAILTO=user1@nowhere.org,user2@somewhere.org
0 2 1-10 * * du -h --max-depth=1 /
```

This line is and example of running a cron job every month, on Mondays whose dates are between 15-21. This means the third Monday only of the month at 4 a.m.

```
0    4 15-21 * 1 /command
```

8.Patrol

8.1Filesystem monitoring

Patrol automatically monitors all mounted filesystems . The SA will be notified on SA owned or OS filesystems, the application is responsible for identifying their filesystems for notification. For filesystems that belong to an application, Patrol can direct notification of alerts to the application team. Only by identifying the filesystem on the AR1 form can Patrol setup the change in who receives notification when the filesystem reaches a near full capacity. All monitors have a set of alarm levels known as thresholds to determine when an alert should occur. The standard threshold for Filesystem Capacity, (known as FSCapacity), **Alarm 1 (warning)** is **96% to 98% Alarm 2** (**alarm**) is **98% to 100%**. All thresholds can be adjusted up or down depending on the needs of the application. On the AR1 form you can state what the thresholds should be changed to if other than the standard. Please make sure it is a mounted filesystem that you want monitored for this section. Using the UNIX "**df** " command you can find out if the filesystem you want Patrol to monitor is a mounted filesystem. When you perform the **df** command, the last column to the right will tell you which device the filesystem is mounted on. Since Patrol automatically monitors all mounted filesystems the only setup required is for you to list the filesystem and desired Alarm Levels (thresholds) on the AR1 form.

8.2 Logfile monitoring

Patrol's AdvLog KM performs Logfile monitoring. Logfiles that have error messages written to it rather than all messages generated by an application are candidates for using the AdvLog KM. The AdvLog KM monitors near real-time messages written to one or more logfiles. Any character string written to a logfile can be detected by Patrol and notification sent.

AdvLog has a duplication message algorithm that can be enabled to ensure that the application team is not overwhelmed with identical messages. AdvLog can monitor the logfile size and logfile rate of growth. AdvLog can be configured to look for "x" number of messages written to a logfile within "x" number of polling intervals before the AdvLog Instance enters an alarm state. An AdvLog Instance is created for each logfile being monitored.

8.3 Process monitoring

Patrol can monitor application processes using the DAEMON KM. Patrol can be configured to watch for one active presence of a process or multiple presence of a process. Also, the DAEMON KM has the ability to restart a failed process if given permission to execute the start command for the failed process.

Shell script monitoring

There are cases when monitoring of a component cannot be handled by standard Patrol KMs. A shell script may be used to check the condition of the component. A return code of zero indicates that the monitored resource is OK. A return code other than zero indicates that the monitored resource is NOT OK. To use the STATPROC KM the shell script must be executable by the patrol id

9. Server

In the context of client-server architecture, a server is a computer program running to serve the requests of other programs, the "clients". Thus, the "server" performs some computational task on behalf of "clients". The clients either run on the same computer or connect through the network.

9.1 TYPES OF SERVERS

There are currently 6000 servers deployed by AT&T for various purposes

There are basically 3 types of servers 4th type comprises of the 3 basic servers.

- Production server
- Test server
- Development server
- Expired server(can be any of above servers)

1. Production servers : These servers are generally business related servers and in terms of ticket handling are the most critical ones . Generally applications of these servers comprise of finance management and sales support.

   As these servers run mission critical services any problem in these servers means a great loss to the company so generally these servers are clustered.

   If any ticket concerning these servers arrive it has to be handled with extreme care and one thing should be kept in mind that these servers can never be rebooted without prior notification to the client company.

2. Test servers :These servers are for testing new applications. After proper testing these applications are installed in production or development servers .

   These servers are itself promoted to production servers after sufficient examination.

3. Development servers : These servers are used to develop new applications.

4. Expired servers :Expired server can be any of the above mentioned 3 servers. These are those servers whose services have been shifted to other servers but they are still running.

Tickets of these servers are not mission critical and can be handled after handling other tickets and these tickets are generally helpful for new comers.

9.2 Accessing the servers

Managing distributed servers from a remote location is mandatory in today's business environment. IT administrators must easily and effectively manage servers in secure data centers or in locations that have no administrative IT staff. Such scenarios require remotely performing all server management operations and responding to server-down situations.

Remote-access capabilities help to improve system administrator productivity and overall server availability by reducing administrator visits to the system and by allowing some operations on groups of systems instead of individual devices.

There were basically three ways to access the AT&T servers from remote location.

1. Putty
2. Amity
3. Web Interface (DRAC,ILO etc.)

9.2.1 Putty

Putty is a free implementation of Telnet and SSH for Win32 and UNIX platforms, along with an xterm terminal emulator. It is written and maintained primarily by Simon Tatham.

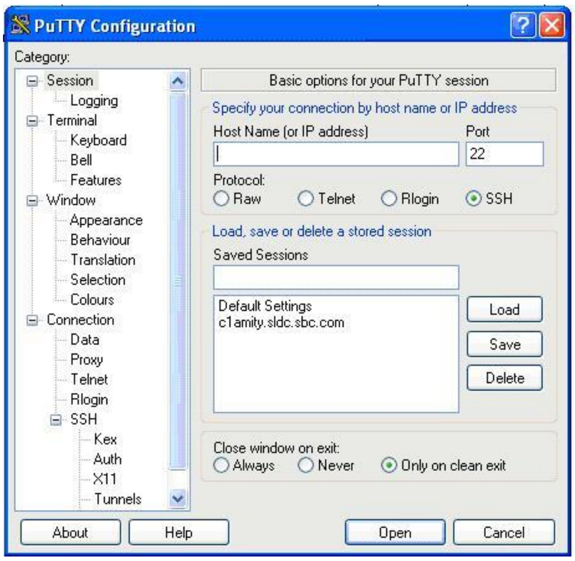
Secure Shell (**SSH**) is a network protocol (Application layer) for secure data communication, remote shell services or command execution and other secure network services between two networked computers that it connects via a secure channel over an insecure network: a server and a client (running SSH server and SSH client programs, respectively).

Putty is a client program for the SSH, Telnet and Rlogin network protocols. These protocols are all used to run a remote session on a computer, over a network. Putty implements the client end of that session: the end at which the session is displayed, rather than the end at which it runs.

When you run Putty on a Windows machine, and tell it to connect to (for example) a UNIX machine, Putty opens a window. Then, anything you type into that window is sent straight to the UNIX machine, and everything the UNIX machine sends back is displayed in the window. So you can work on the UNIX machine as if you were sitting at its console, while actually sitting somewhere else.

Some features of putty are:

- The storing of hosts and preferences for later use.
- Control over the SSH encryption key and protocol version.
- Command-line SCP and SFTP clients, called "pscp" and "psftp" respectively.
- Ipv6 support.



Host Name (or IP address)

Port: 22

Protocol: ○ Raw ○ Telnet ○ Rlogin ● SSH

Saved Sessions:
Default Settings
c1amity.sldc.sbc.com

Close window on exit: ○ Always ○ Never ● Only on clean exit

9.2.2 Amity

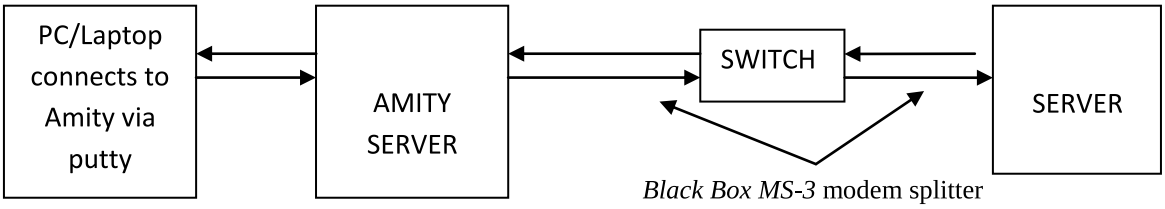
AMITY runs as an application on the *IBM RISC/6000* System, using IBM's *Advanced Interactive Executive Operating System (AIX) and X-Windows*. It enables users to monitor and access many computer systems from a centralized location. The Amity application does not "reside" on the monitored system. It is physically connected to the monitored system, which may be done in a number of ways.

Typically an asynchronous connection is made through a . The cable that normally connects a server to its system console is connected to the Master port on the MS-3 splitter. Another cable connects the MS-3 to the system console and a third cable connects the MS-3 to the Amity system. System messages are then received at both the system console and the Amity application. Since the cables allow two-way communications with the monitored system, system responses can be returned from either the system console or from Amity, either as automatic responses or as manually entered commands.

As system messages are received by Amity from each monitored system, Amity stores then analyzes message text to determine if a response is warranted. Amity uses a separate database file for each monitored system to analyze messages received from each system.

Each database file contains strings of text for which Amity is looking and actions designating how Amity should respond to the received message if it should match an entry in the database file. Database actions include alphanumeric paging, audible voice messages, issuing commands back to the monitored system, e-mail, and messages to SPG (Single Pane of Glass).

Use Amity to connect to actual server. Connect to Amity via Putty using hostname or IP address. Next, from Amity, connect to the server's console. We can connect directly to the server via SSH or Telnet using Putty but when we reboot (as a part of verification), the server connection will be lost.


PC/Laptop connects to Amity via putty
AMITY SERVER
SWITCH
SERVER
Black Box MS-3 modem splitter

Using Amity helps us to overcome the problem of lost connection while reboot. The connection of server is done on the console and not at the network port.

9.2.2.1 Amity Command Reference

- **looklog** command

Purpose

Provides access to the console logfile of an AMITY connected system.

Syntax

looklog *systemname* [*Mmmdd*]

Description

The looklog command is an text-based interface to any of the stored console logfiles of AMITY connected systems. If the system name is valid and you have authorization to access the system you requested, your session screen will be cleared and today's logfile is presented with the UNIX pg command. The key sequence used to end a pg session before reaching the end of file (EOF) is CTRL/C.

The looklog command, entered with a valid system name and a valid date as its argument, will behave as above with the logfile of the requested date.

The looklog command does not access systems connected to other AMITY nodes. If you have Xwindows running on your desktop, use the **logscan** command to access systems connected to other AMITY nodes. If you do not have Xwindows available, use the **where** command to determine to which AMITY node your desired system is connected, login to that AMITY node, and then run the looklog command on that node.

Examples

1. To access a system console for which you are authorized, enter:

```
looklog systemA
```

2. To use the looklog command with an argument for a date other than today, enter:

```
looklog systemA Jan01
```

This will present the logfile for *systemA* for January 01.

Files

/home/amity/bin/looklog contains the looklog command.

- **one** command

Purpose

Provides access to the console of an AMITY connected system.

Syntax

one [*systemname*]

Description

The **one** command, entered without arguments, displays the list of systems connected to the local AMITY node and asks you to select one. Type the name of the system to which you want console access and press the Enter key. If the system name is valid and you have authorization to access the system you requested, your session screen will be cleared and you will be asked to re-enter the system name. If the two entries match, the following will be displayed: a notice indicating if the session is read-only, an indicator of the key sequence used to end this session, and the last 50 lines of the logfile. A response other than a valid system name returns an error message and the AMITY system prompt.

If at least one of the five existing logical connections to the console is available, you will be given read/write access to the console. The key sequence normally used to end a read/write session is CTRL/A. If all five of the logical connections to the console are in use, you will be given read-only access to the console. The key sequence used to end a read only session is CTRL/C.

The **one** command, entered with a valid system name as its argument, will behave as described above, bypassing the listing of and the initial prompt for the desired system.

The **one** command does not access systems connected to other AMITY nodes. If you have Xwindows running on your desktop, use the **console** command to access systems connected to other AMITY nodes. If you do not have Xwindows available, use the **where** command to determine to which AMITY node your desired system is connected, login to that AMITY node, and then run the **one** command on that node.

Examples

1. To access a system console for which you are authorized, enter:

```
one
```

This displays the message:

```
systemA     systemB     ...     system1     system2

Enter the system name: systemA
```

You must now re-type the system name that you want to access.
Please enter the system name: systemA

Typing a system name gives you read/write or read-only access. You can press the Enter key without typing a system name to quit the command. This will display the message:

You did not enter a system name.

- **where** command

Purpose

Displays information about an AMITY node and system(s) connected to it.

Syntax

where *systemname*

Description

The **where** command, entered with a valid system name as its argument, will present one of two types of display, depending on whether *systemname* is an AMITY node or an AMITY connected system. If *systemname* is one of the AMITY nodes (c1amity, c2amity, s1amity, s2amity, txamity, tyamity), the output is a list of all systems connected to that AMITY node. If *systemname* is an AMITY connected system, the output displays to which AMITY node *systemname* is connected.

If you have Xwindows running on your desktop, you can use the **console** and **logscan** commands to access system consoles and logfiles connected to any AMITY node, no matter to which AMITY node you are logged in. However, if you do not have Xwindows available, you must login to the AMITY node local to the system in which you are interested and use the **one** and **looklog** commands. Use the **where** command to determine to which AMITY node your desired system is connected.

Examples

1. To access a system console for which you are authorized, enter:

where xnamity

This displays the message:

```
systemA        systemB        ...        system1        system2
```

and the AMITY system prompt.

2. To use the **where** command with an argument to determine a connected system's local AMITY node, enter:

```
where systemA
```

This displays the message:

```
systemA is running on s1amity
```

and the AMITY system prompt.

3. An error message is generated if the **where** command is invoked without any arguments

```
where
Usage: where system_name
```

Output is a statement about which AMITY system is running the argument

Files

/home/amity/bin/where contains the **where** command.

9.2.3 Web Interface

Access using web interface is generally done when the server is not responding to putty and even Amity. It happens in the case when server is down and is to be rebooted to get its services up.

Paste the RIB (Remote insight board) IP address in Internet explorer.

All vendors who supply servers install the interface cards in their servers. During my training period I came across 3 interface cards from

- Dell
- HP (Hewlett Packard)
- IBM

A special property of these interface cards is that they use out-of-band or lights out management.

Out of band management

In computing, out-of-band management (sometimes called **lights-out management** or **LOM**) involves the use of a dedicated management channel for device maintenance. It allows a system administrator to monitor and manage servers and other network equipment by remote control regardless of whether the machine is powered on, or if an operating system is installed or functional.

In band management

In band management like VNC(virtual network circuit) or SSH is based on software that must be installed on the remote system being managed and only works after the operating system has been booted. This solution may be cheaper, but it does not allow access to BIOS settings, or the reinstallation of the operating system and cannot be used to fix problems that prevent the system from booting.

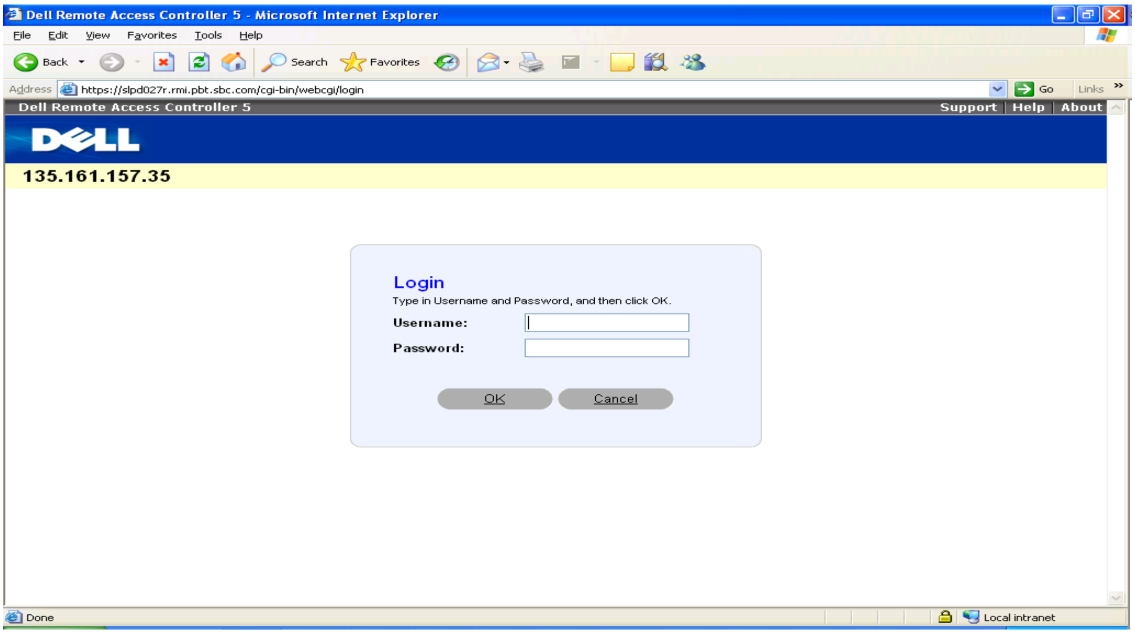
9.3.1 Dell DRAC (Dell Remote Access Controller)

The Dell Remote Access Controller or DRAC , an interface card from Dell , provides out-of-band management facilities. The controller has its own processor, memory, network connection, and access to the system bus. Key features include power management, virtual media access and remote console capabilities, all available through a supported web browser or command line interface.

The Dell remote-access architecture consists of hardware and software components that allow administrators to do the following:

- Access a server after a server failure, power outage, or loss of a network connection (using a network interface card (NIC) or modem)
- Remotely view a server's internal event logs and power-on self test (POST) codes for diagnostic purposes
- Manage servers in multiple locations from a remote console
- Manage servers by redirecting the console output to a remote console (graphic and text)

- Perform an orderly shutdown of a server for maintenance tasks
- Diagnose a server failure and restart the server


Dell Remote Access Controller 5 - Microsoft Internet Explorer
File Edit View Favorites Tools Help
Back Search Favorites
Address Go Links
Dell Remote Access Controller 5
Support | Help | About
DELL
Done
Local intranet

https://slpd027r.rmi.pbt.sbc.com/cgi-bin/webcgi/login

135.161.157.35

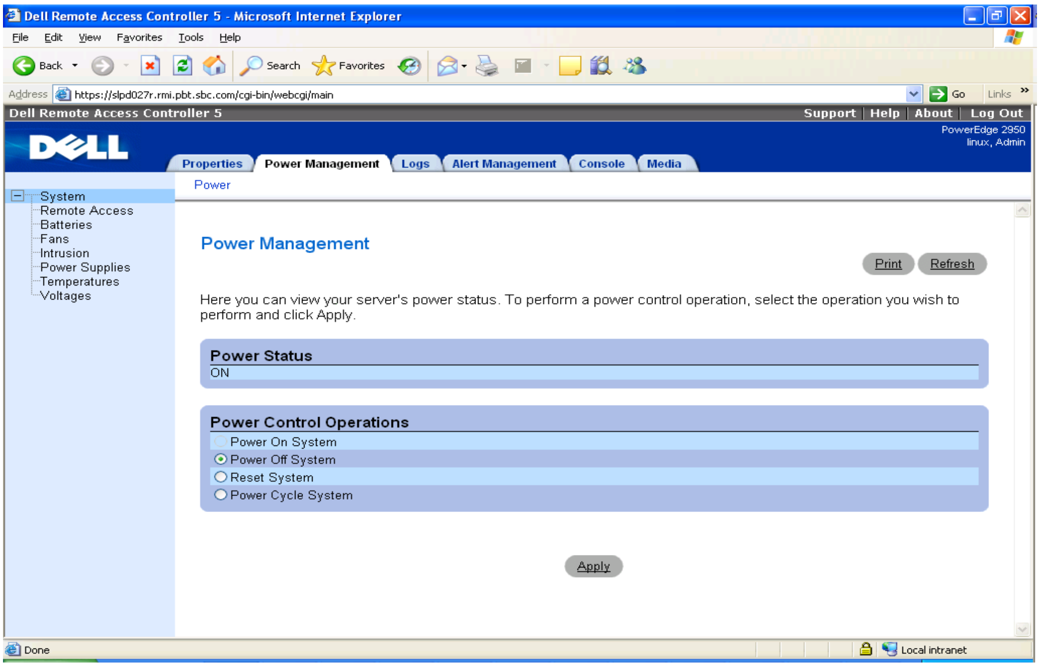
Login

Type in Username and Password, and then click OK.

Username:

Password:

This is the interface which comes after you enter the RIB IP. Here the SA write the username and RCA password.



https://slpd027r.rmi.pbt.sbc.com/cgi-bin/webcgi/main

PowerEdge 2950
linux, Admin

Power Management

Here you can view your server's power status. To perform a power control operation, select the operation you wish to perform and click Apply.

Power Status

ON

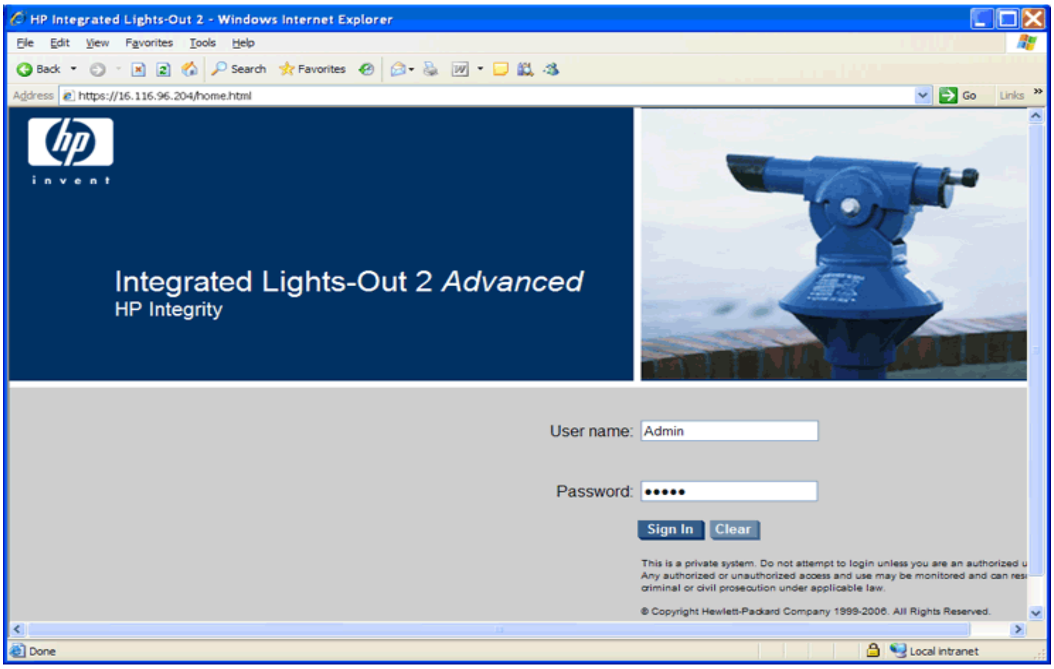
Power Control Operations

Power On System
● Power Off System
○ Reset System
○ Power Cycle System

From here the server can be rebooted.

9.3.2 HP iLO (Integrated Lights Out)

Integrated Lights-Out,or iLO, is an embedded server management technology exclusive to Hewlett-Packard but similar in functionality to the Lights out management (LOM) technology of other vendors, for example Dell DRAC.



https://16.116.96.204/home.html

User name: Admin

Password: •••••

This is a private system. Do not attempt to login unless you are an authorized u
Any authorized or unauthorized access and use may be monitored and can res
criminal or civil prosecution under applicable law.

© Copyright Hewlett-Packard Company 1999-2006. All Rights Reserved.

10. Tickets

There are two types of tickets in general

1. Internal Engagement ticket : This kind of ticket is created by the EMOC team which tries to solve it if possible else it sends it to the USH LINUX team.

Engagement ticket can never be closed by the USH team. It is put in "Ready to close state and sent back EMOC team which closes it.

Engagement tickets are generally due to alerts generated by PATROL agents.

2. Main ticket(User ticket): This kind of tickets are created by the end users itself for the concerned teams.

   These tickets can be closed after the work has been done on the problem mentioned in the ticket.

10.1 Example of Some Tickets

Problem -1

On a server "xyz" swap space is full. Check out.

***What to* do**

Check the virtual memory using command-

Ps – e - -sort - vsz - o comm, pmem,user,rsz,vsz,pid |head -11

From this command we will get to know which user is using the maximum space

Cron related tickets

Problem -2

On server "xyz" 100 cron processes are running. Kill the processes.

What to do

Login to the respective server using PUTTY.

Kill all the cron processes using kill -9 command.

# kill -9 pidno.

"-9" ensures execution.

Issue

Everyday on the same server same cron process are repeatedly generated.

Root Cause Analysis

Application owners edit the crontab file regularly so that they can schedule a particular process for themselves. Now if many people have done so than it may be that at the same time there are many process which come into existence terms it's performance decreases.

Now in this case a ticket may be generated by a person who is trying to use the server or is trying to run its application or a person whose application was already running and now has been affected may create a ticket. Or even a patrol agent which is monitoring the maximumnumber of processes that can be run simultaneously may generate an alert for EMOC team which in turn generates a ticket for USH-SA team.

Crontab File

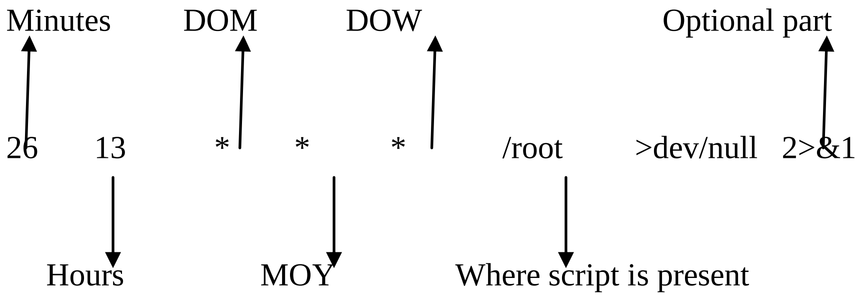
It is simple text file where the schedule of a particular job has to mentioned.

It is present in

/etc/crontab

/usr/bin/crontab

It's format is



Minutes DOM DOW Optional part

26 13 * * * /root >dev/null 2>&1

Hours MOY Where script is present

DOM – Date of Month.

MOY – Month of Year.

DOW – Day of week. (sun = 0,7; mon = 1 and so on)

Problem -3

On a veristas cluster server "xyz" a process of sending timestamp is running many times.Kill it and do root cause analyses.

What to do

Simple solution to this problem is killing the processes but as it is asked to do root cause analyses we have to look for the reason that why is it happening so.

Basically sending Time stamp is property of veritas clusters to check whether other servers in the cluster are working or not. So every server in a cluster sends Time stamp to other server which in turn acknowledges it by sending another timestamp back. Now this process takes place every 15 minutes.

In this case what happened is that a server is not acknowledging the server which has sent timestamp to it. It means that either server is down but we check that the server is not down by logging in it through putty.

As this process takes place every fifteen minutes so by the end of the day around 100 processes are made.

Veritas cluster are generally connected ethernet LLT cables and they contact with each other through these cables. So only one possibility is left that there is problem with these cables.

In this we will engage a mail for the deployment team of the onshore department. In this mail we will specify the location of the server from the IEDS database.

Problem -4

On a server "xyz" a certain script should run on this instance.

What to do

Just edit the Crontab file using any text editor and add the information in the formatted sated above.

Note

The script to be run are always written by on-shore team of AT&T. No employee of tech Mahindra can ever run any script of their own.

Problem -5

Account created by UAM team. Will be active tomorrow. But need it immediately.

Description - UAM team creates an account for new employee. Assign it various permissions. Now what UAM team does is it runs a script which creates the account in every server of the company. But the team runs this script only once a day for all the new employees who have joined in the day

What to do

Check the crontab file for the particular script its Location and run the script manually so that the account can be created immediately.

Problem -6

On a server "xyz" an application needs to be run. Set the uplimit for the number of files it can access.

Description - Any application when run on a server may require to open a certain number of files. Now these files should be finite in number and if the process tries to open extra files than the limit that has been set then it may affect the performance of the server .An alert is generated by patrol agent.

What to do

Login to the given server and edit **the etc/security/limit.conf** file using the vi editor and enter the application with its respective hard and soft number.

```
Appname    soft    nofile    65536

Appname    hard   nofile    65536
```

Problem -7

On a server “xyz” /var filesystem is 96%.

The /var filesystem

The /var contains data that is changed when the system is running normally. It is specific for each system, i.e., not shared over the network with other computers.

/var/log

Log files from various programs, especially login(/var/log/wtmp , which logs all logins and logouts into the system) and syslog (/var/log/messages , where all kernel and system program message are usually stored). Files in /var/log can often grow indefinitely, and may require cleaning at regular intervals.

/var/mail

This is the FHS approved location for user mailbox files. Depending on how far your distribution has gone towards FHS compliance, these files may still be held in /var/spool/mail .

/var/tmp

Temporary files that are large or that need to exist for a longer time than what is allowed for/tmp . (Although the system administrator might not allow very old files in /var/tmp either.)

What to do

These are 3 subdirectories under the /var directory which take most of the space.

```
 # du-sh *|grep M

# lsof log

# gzip -9 filename – to compress the files.
```

Problem -8

Install or uninstall the following package on the specified server

What to do

To view the package use command

# rpm –qa (package-name).

If Red Hat Enterprise linux release level 3,4 then use

# rpm –ivh (package-name).

If Red Hat Enterprise linux release level 5,6 then use

# yum –install (package-name)

To uninstall a package use

# rpm –e (package-name)

To upgrade a package use

# rpm –uvh (package-name) .

# Yum –upgrade(package-name)

Problem -9

Need to set system clock to a certain Timezone.

What to do

Link the */usr/share/zoneinfo/UTC to /etc/localtime*

/usr/share/zoneinfo file contains all the zone times.

/etc/localtime is a file from where the system reads the localtime.

Use commands

# cd /usr/share/zoneinfo

# ln –f /usr/share/zoneinfo/UTC /etc/localtime

ln command with '*–f* ' creates link without questioning the user.

Problem -10

On a server "xyz" Create swap partition of given size.

What to do

There are many ways to create a swap partition but you need to have root permissions to make it.

Free command displays the swap space. free -k shows the output in KB.

Swapon command with option -s, displays the current swap space in

1.*fdisk command*

#fdisk /dev/hda

New:n

Size :Mention the size of swap partion to be created.

#mkswap /dev/hda7.

#swapon /dev/hda7.

swapon command enables the swap space just created .

Problem -11

Server appears down please check.

What to do

1. Try to Ping the sever.
2. Then access it through Amity.
3. Go to */etc/ssh/sshd_conf.*
4. Restart the sshd service using command
   #service sshd restart
5. If still not able to ping the server then check Ethernet ports using command
   #ifconfig
   #ifup eth0
6. If not able to ping use *mii –tool*
   (mii –tool shows connectivity of every root interface)

At last notify that it is a hardware problem.

11. File locking mechanism

11.1 Problem statement

A file named /etc/passwd has to be locked via a command for 30 seconds and the user must get control after the file is locked i.e the user gets the terminal back.

For unlocking there should be two ways

- First the file unlocks itself after 30 seconds.
- Second, the user issues an unlock command.

11.2 Problem description

etc/passwd file

etc/passwd file stores essential information, which is required during login i.e. user account information. /etc/passwd is a text file, that contains a list of the system's accounts, giving for each account some useful information like user ID, group ID, home directory, shell, etc. It should have general read permission as many utilities, like ls use it to map user IDs to user names, but write access only for the superuser (root).

The /etc/passwd contains one entry per line for each user (or user account) of the system. All fields are separated by a colon (:) symbol. Total seven fields as follows.

Generally, passwd file entry looks as follows :

Jsmith : x: 1001 : 1000 : Joe Smith,Room 1007,email : /home/jsmith : /bin/sh

1. **Username**: It is used when user logs in. It should be between 1 and 32 characters in length.
2. **Password**: An x character indicates that encrypted password is stored in /etc/shadow file.
3. **User ID (UID)**: Each user must be assigned a user ID (UID). UID 0 (zero) is reserved for root and UIDs 1-99 are reserved for other predefined accounts. Further UID 100-999 are reserved by system for administrative and system accounts/groups.
4. **Group ID (GID)**: The primary group ID (stored in /etc/group file)
5. **User ID Info**: The comment field. It allow you to add extra information about the users such as user's full name, phone number etc. This field use by finger command.
6. **Home directory**: The absolute path to the directory the user will be in when they log in. If this directory does not exists then users directory becomes /
7. **Command/shell**: The absolute path of a command or shell (/bin/bash). Typically, this is a shell. Please note that it does not have to be a shell.

11.3 Pseudo code

C Libraries included:

- stdlib.h-
- shadow.h
- unistd.h
- errno.h
- signal.h
- string.h
- stdio.h
- sys/fcntl.h
- sys/time.h

Macro’s defined

- LOCKFILE “/tmp/passwd_lck
- OPIONS “:luhv”
- TIMEOUT 30

SIGNAL HANDLERS USED:

Return type:void

Handler: ALARMhandler(int sig)

SET timeout=0;

Handler: USR1handler(int sig)

SET timeout=0;

FUNCTONS USED:

Function :checkpreviouslock(int status, char **argv)

Return type: void

Definition:

```
Declare int res=-1, oldpid.
IF (LOCKFILE is accessed)
    CALL getoldpid();
     WRITE “ps-ef| grep oldpid  to global variable ‘cmd’.    //use sprint function
```

```
        RUN system command.                     // use System function
        IF (successful)
            IF (!status)
                 WRITE "previous lock exist pid=oldpid"
                EXIT
            ELSE
                KILL old process with pid=oldpid
                REMOVE LOCKFILE
            END – IF ELSE
        ELSE
            REMOVE LOCKFILE
        END - IF ELSE
END IF
```

System defined functions used:

1.access ()

Synopsis

#include <unistd.h>

int access(const char *pathname, int mode);

Description

access() checks whether the calling process can access the file *pathname*. If *pathname*is a symbolic link, it is dereferenced.

The *mode* specifies the accessibility **check**(s) to be performed, and is either the value**F_OK**, or a mask consisting of the bitwise OR of one or more of **R_OK**, **W_OK**, and**X_OK**. **F_OK** tests for the existence of the file. **R_OK**, **W_OK**, and **X_OK** test whether the file exists and grants read, write, and execute permissions, respectively.

2. system(cmd)

Synopsis

```
#include <stdlib.h>

int system(const char *string);
```

Description

system() executes a command specified in string by calling /bin/sh -c string, and returns after the command has been completed. During execution of the command, SIGCHLD will be blocked, and SIGINT and SIGQUIT will be ignored.

RETURN VALUE

The value returned is -1 on error (e.g. fork failed), and the return status of the command otherwise.

3.kill

Synopsis

```
#include <sys/types.h>
#include <signal.h>

int kill( pid_t pid, int sig );
```

Description

The **kill()** function sends a signal to a process or process group specified by *pid*. The signal to be sent is specified by *sig* and is either 0 or one of the signals from the list in the <**sys/signal.h**> header file.

The process sending the signal must have appropriate authority to the receiving process or processes. The **kill()** function is successful if the process has permission to send the signal *sig* to any of the processes specified by *pid*. If **kill()** is not successful, no signal is sent.

4. remove

Synopsis

```
#include<stdio.h>

int remove ( const char * filename );
```

Description

Deletes the file whose name is specified in *filename*.
This is an operation performed directly on a file; No streams are involved in the operation.

If the file is successfully deleted, a zero value is returned.

Function : lock(char)

Return type : void

Definition :

```
Declare FILE *fp, pid_t pid ,sigset_t set
CALL check previouslock(0,argv)
FORK the process
IF(child process)
        CALL lckpwdf()
        IF (error)
                WRITE “call to lckpwdf failed”
        ENDIF
        CREATE LOCKFILE
        WRITE pid of child process inLOCKFILE
        BLOCK signals SIGINT,SIGTERM,SIGSTOP
        DECLARE signal(SIGALRM,ALARMhandler)
                      signal(SIGUSR1,USRhandler)
        CALL alarm for 30 timeout
        WHILE(timeout)
                SLEEP for 1 sec.
        END WHILE
        UNBLOCK signals SIGINT, SIGTERM, SIGSTOP
        CALL ulckpwdf()
        IF(ERROR)
                WRITE call to ulckpwdf() failed.
        END IF
        CALL alarm for 0 sec
```

```
ELSE IF (parent process)
    WRITE “trying to get lock”
    WHILE(1)
        IF (LOCKFILE Exist)
            BREAK
        ELSE
            WRITE “.”
            SLEEP for 1 second
        END- IF ELSE
    END WHILE
END IF ELSE
```

Data types used

1. pid_t

The ***pid_t*** data type represents process IDs.

2. sigset_t

The sigset_t data type is used to represent a signal set. Internally, it may be implemented as either an integer or structure type.

System functions used

1. fork()

Synposis

```
#include<unistd>

int c=fork()
```

The fork() system call takes no argument and returns a process ID, which is usually an integer value. The returned process ID is of the type **pid_t**, which has been defined in the header file,**sys/types.h**.

The purpose of fork() system call is to create a new process, which becomes the child process of caller, after which both, the parent and child processes, will execute the code following the fork() system call. Hence, it is important to distinguish between parent and child process. This can be done by testing the return value of fork() system call.

- If fork() returns a negative value, it indicates that the creation of the process was unsuccessful.
- fork() returns a zero to the newly created child process.
- fork() returns a positive value, the process ID of the child process, to the parent.

2. lckpwdf()

Synopsis

#include<shadow.h>

int retcode=lckpwdf()

Description

The **lckpwdf**() function is intended to protect against multiple simultaneous accesses of the shadow password database. It tries to acquire a lock, and returns 0 on success, or -1 on failure (lock not obtained within 15 seconds).

3. ulckpwdf()

Synopsis

#include<shadow.h>

int retcode=ulckpwdf()

Description

The **ulckpwdf**() function releases the lock again. Note that there is no protection against direct access of the shadow password file. Only programs that use **lckpwdf**() will notice the lock.

4. sigemptyset

Synopsis

#include<signal.h>

Sigemptyset(sigset_t set)

Description

This function initializes the signal set *set* to exclude all of the defined signals. It always returns 0.

5. sigaddset

Syonpsis

#include<signal.h>

Sigaddset(sigset_t set, int signum)

Description

This function initializes the signal set *set* to exclude all of the defined signals. It always returns 0.

6. sigprocmask

Synopsis

#include<signal.h>

Sigprocmask(int how, sigset_t *set, sigset_t *oset)

Description

In a single-threaded process, the sigprocmask*()* function allows the calling process to examine or change (or both) the signal mask of the calling thread. If the argument *set* is not a null pointer, it points to a set of signals to be used to change the currently blocked set.

The argument *how* indicates the way in which the set is changed, and consists of one of the following values:

SIG_BLOCK

The resulting set will be the union of the current set and the signal set pointed to by *set*.

SIG_SETMASK

The resulting set will be the signal set pointed to by *set*.

SIG_UNBLOCK

The resulting set will be the intersection of the current set and the complement of the signal set pointed to by *set*.

If the argument *oset* is not a null pointer, the previous mask is stored in the location pointed to by *oset*. If *set* is a null pointer, the value of the argument *how* is not significant and the process' signal mask is unchanged; thus the call can be used to enquire about currently blocked signals.

If there are any pending unblocked signals after the call to *sigprocmask()*, at least one of those signals will be delivered before the call to *sigprocmask()* returns.

It is not possible to block those signals which cannot be ignored. This is enforced by the system without causing an error to be indicated.

7. alarm

Synopsis

#include<time.h>

alarm(int sec)

The alarm function sets the real-time timer to expire in seconds seconds. If you want to cancel any existing alarm, you can do this by calling alarm with a seconds argument of zero.

The return value indicates how many seconds remain before the previous alarm would have been sent. If there is no previous alarm, alarm returns zero.

Function :getoldpid()

Return type : int

Definition:

```
DECLARE FILE *fp, int oldpid.

OPEN file “LOCKFILE”

IF(successful)

        READ pid to oldpid;

        CLOSE file

END IF

RETURN oldpid.
```

Function: unlock(char **argv)

Return type :void

Definition:

```
CALL checkpreviouslock(1,argv)
```

Function : main(int argc, char argv)

Return type : int

```
DECLARE int c, int timeout
```

```
SET timeout=TIMEOUT
INITIALIZE memory to global variable “cmd”        //use memset
WHILE(c=getopt(argc,argv,OPTIONS)!=-1)
        Switch(c)
        Case ‘l’ : CALL lock(argv); break;
        Case ‘u’ : CALL unlock(); break;
        Case ‘v’ : WRITE version 1.0.0 15/4/2010\nAdept lock for passwd file break;
        Case ‘h’ : CALL usage()break;
return 0;
```