



SOX Compliance for AutoSys

iXp: Simplified Security and Management Solution

Overview

Application security is crucial to Sarbanes-Oxley compliance. Any system that stores or manages financial data and processing is required to be compliant with the SOX guidelines. In addition, certain reporting needs have to be met. Lastly, vulnerabilities have to be detected and fixed before they can be exploited.

From AutoSys, companies manage critical processes that cover various business areas. These business areas are exposed to danger if security, reporting, and auditing for the AutoSys environment is not enforced.

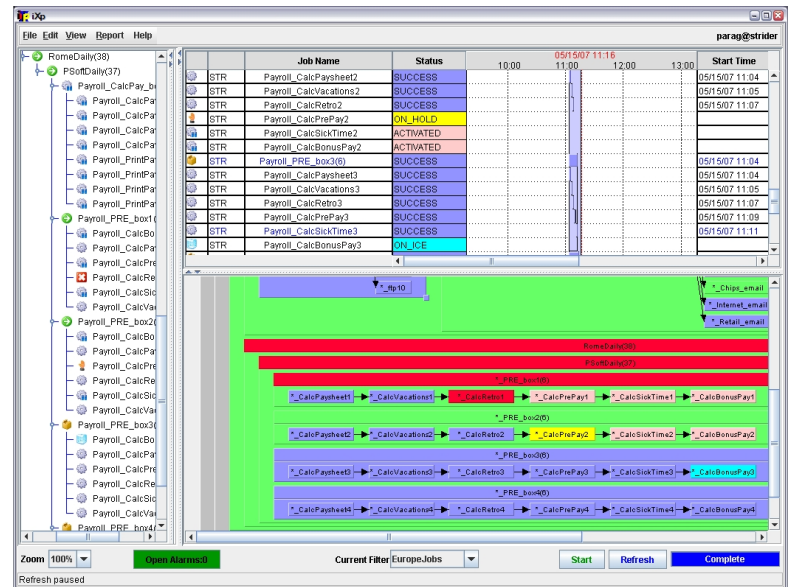
iXp is a 3-tiered Java-based application that offers a complete security, management, reporting, and auditing solution for web-based and command-line access to AutoSys. By providing a comprehensive security and auditing model, you can enforce the required access control and reporting policies for the entire AutoSys environment.

Key Features

Client Interfaces

Web-based GUI: The iXp GUI client can be launched from a web-browser. It enables users to monitor, update, control, forecast, simulate, report on, and print AutoSys job streams across multiple instances. The iXp GUI provides a variety of graphical views, including Job Flow, Gantt, Tree, and Console views. Since the GUI is web-based, no software installation is required.

Thin Client CLI: iXp also provides command line utilities that provide the capability to view, report, control, create, update, delete AutoSys jobs and global variables. The commands can be executed from any client system and do not require a local AutoSys software installation. The CLI is counterpart to standard AutoSys commands for the same function (e.g. *autorep*, *sendevent*, *jil*). The commands support the same input and output format as their AutoSys counterparts.



The web-based iXp GUI enables users to view the real-time information for AutoSys job streams in a variety of graphical views.

Customized View creation: Users and administrators can create unlimited numbers of views that show different job streams based on Job attributes. You can specify multiple values for each attribute, including wildcards, to create views specific to your business areas.

User Authentication

Single Sign-On (SSO): Each user has to be successfully authenticated before they can use the GUI or CLI clients. iXp supports Active Directory / NT Domain, and OS-based SSO authentication. The iXp Administrators can provide a granular list of valid Domains. Users that are not logged on to those domains will be unable to launch the GUI or the CLI.

Host and IP Address Validation: Administrators can specify a wild-carded list or range of IP Addresses and host names. Users that have successfully authenticated to valid Domains will be denied access to the iXp GUI or CLI if the client machine does not match the list of IP Addresses and host names.

User Authorizations

AutoSys Job Privileges: Users have to be assigned authorization policies in order for them to access iXp. iXp supports *Job Read*, *Job Control*, *Job Insert*, *Job Update*, *Owner List* and *Job Override* privileges. For each privilege, Administrators can assign authorized list of jobs (and owner names for *Owner List*) to each user. These lists are based on multiple values, including wildcards, for different job attributes. Values can be provided for job attributes such as *name*, *machine*, *owner* etc. Each user can be authorized with multiple lists for each privilege.

AutoSys Alarm Privileges: Administrators can also assign authorizations to enable users to Acknowledge, or Close AutoSys Alarms. Same lists as those used for Job Privileges can be used.

The iXp Authorization model is common to both the GUI and CLI. Users issuing events from the GUI or using the *sendevent* command will pass through the same security model.

Other Features

Forecasting and Simulation: iXp provides a real-time forecasting engine that enables users to view upcoming job runs. Users can also run a forecast for any day in the future. The forecast can be run from the GUI or the CLI, and the output of the forecast can be viewed in the GUI or saved as a HTML, CSV, or text file. Users can simulate “what-if” scenarios by providing runtime, status, and sendevent values.

Historical Reporting: Users can create customized report definitions and generate HTML, CSV, text reports against historical AutoSys job runs data from the GUI or the CLI. Users can also view graphs and generate reports about AutoSys Performance, including per minute Total Latency, Events Processed, Job Starts and Ends.

Key Benefits

Simplify Total AutoSys Security: You can secure access to AutoSys and generate audit trails of all activity

CONTACTS

PGTI

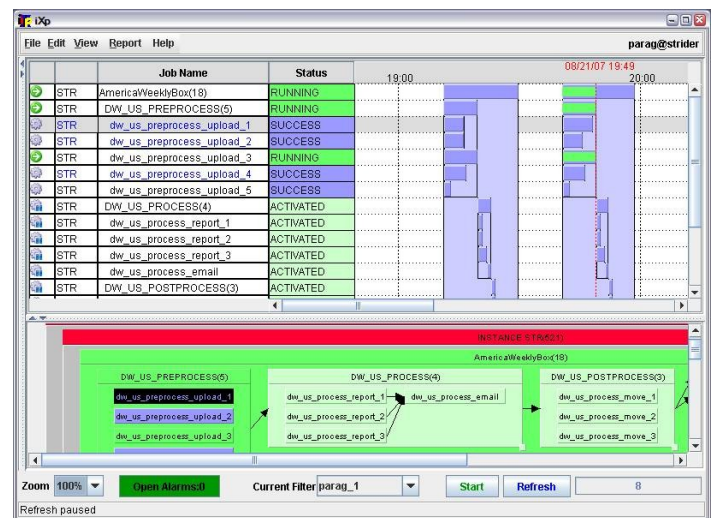
8753 Yates Drive, Suite 200, Westminster, CO 80031

Ph: (303) 301-2667 Email: info@pgti.com

URL: www.pgti.com

performed by users from the Web GUI or the CLI. The iXp Security model is common across the GUI and the CLI, and does not need any additional software. This simplifies the maintenance of user authorizations and audit trails. By leveraging the Single Sign-On capability of iXp, no user passwords have to be downloaded or stored.

Increase Service Availability: iXp is a stable and robust product with a small footprint. It has been used in large Production environments since the year 2000. By deploying iXp and its security model, you can provide AutoSys access to large number of users. The stability and scalability of iXp contributes to the availability of your AutoSys environment.



The iXp Gantt View shows live job run information and average run times with indicators for long job run durations..

Single Solution: iXp enables users to perform all AutoSys related activities from the GUI and the CLI. It provides the capability to create and manage AutoSys objects, generate historical and forecast job reports, assign granular authorizations to users, view and monitor Critical Path of jobs, forecast and simulate job stream executions, view and report audit trails of all user actions, monitor cross instance and mainframe job dependencies.

