Stirling&Young

OpenVMS / RMS Dynamic API

SGCO has developed a revolutionary fully functional and industry standard 100% dynamic JSON API for RMS. This will allow real time interaction with any RMS data files for select/read, delete, add and update with the two latter having properly locked rows.

Lightweight:

The major design concern in architecting our API was to be lightweight and efficient. The competing solutions put an unnecessary load onto the host computer resulting in the need to upgrade hardware to compensate. This is turn leads to more issues and a cycle of constant hard to identify problems.

Our API uses only the processing power and memory needed to provide access to RMS. There are no unnecessary spoolers, tracking tables etc. needed to keep the API working as it relies only on the current state of the RMS data in 100% real time. Other commercial solutions either require a service that runs on OpenVMS and a service that runs remotely that must semi-manually be kept in sync and are not truly real time or dynamic or they put an unnecessary load on the OpenVMS host to generate full web content.

Our API consists of only a very small application the runs detached on the OpenVMS OS. It accepts JSON input via posts to a listener that can be assigned to any port, processes that input, performs the requested tasks on the RMS files and returns and necessary response.

Dynamic:

By leveraging RMS indexing in conjunction with the DDL/Data Dictionary for the target data we are able to access the needed data even as the RMS structure is changed. The DDL/Data Dictionary is housed on the remote system and used to dynamically structure queries to RMS as well as the parse the RMS returned records.

Secure:

Our API is by design as secure as possible. The API is not only limited to accept only properly structured JSON input but will only accept input stings that are structurally compatible with the target RMS file. This prevents the API from blindly accepting possibly dangerous input.

Security may also be further enhanced by front ending the API using a second network interface on the host OpenVMS box onto a dedicated LAN with two other entities: A PCI compliant - WAF enabled reverse proxy and a Cisco 5500 series firewall. The Cisco 5500 provides further traffic management as well as the ability to add Cisco Firebox IDS functionality.

It should be noted that this is addressing only the security of the API and that security for the remote applications that manipulate the RMS filesystem via the API are a separate and equally important issue.