Rocket

COMPLIANCE

# Trust Services Principles for Service Organization Controls Reports with Rocket® API

Service Organization Controls (SOC) reports are an effective way for companies to provide assurance to their customers and prospects about the security, availability, confidentiality, integrity, and privacy of the systems they offer. SOC 2 and SOC 3 reports are popular with Software-as-a-Service (SaaS) providers and any company with access to its customers' critical systems and data.

Each Trust Services Principle includes a number of criteria that must be satisfied by the service organization, as well as illustrative controls for how an organization may meet the criteria.

Rocket® API can be a fundamental underpinning of your product or service offering. It has been designed to satisfy all the applicable security requirements to help you achieve SOC certification. Relevant criteria, and the capabilities API offers to meet them, are listed below.

| Criteria | Rocket API Capabilities |
| --- | --- |
| CC5.1<br>Logical access security software, infrastructure, and architectures have been implemented to support (1) identification and authentication of authorized users; (2) restriction of authorized user access to system components, or portions thereof, authorized by management, including hardware, data, software, mobile devices, output, and offline elements; and (3) prevention and detection of unauthorized access. | Rocket API leverages access credentials from the back-end mainframe operating system, and thereby inherits all access rights and restrictions associated with those credentials, including read and write capabilities.<br><br>In addition to access rights controlled through the back-end mainframe, Rocket API provides application-layer security that can further restrict API calls by user, by function, and by data being accessed.<br><br>The Rocket Access and Connectivity Hub (RACH) management interface, which manages the inventory of APIs and deployment to API gateways, enforces granular user access controls that are configurable by each customer.<br><br>RACH uses LDAP authentication to leverage the password controls and other mechanisms that authenticate your users.<br><br>RACH audit logging records all user activity within the application—including uploading and deployment of compiled APIs, as well as administration of the application itself—providing individual accountability for all access and activity. |
| CC5.3<br>Internal and external system users are identified and authenticated when accessing the system components (for example, infrastructure, software, and data). | Rocket API leverages authentication mechanisms already configured within your mainframe environment, inheriting the security policies and configurations you have deployed.<br><br>On top of the authentication security enforced through the mainframe, RACH can add an additional layer of access control through user credentials or tokens. |
| CC5.4<br>Access to data, software, functions, and other IT resources is authorized and is modified or removed based on roles, responsibilities, or the system design and changes to them. | RACH audit logging records all user activity within the application—including administration of the application itself—providing audit evidence of all changes to logical security configurations for review. |
| CC5.6<br>Logical access security measures have been implemented to protect against security, availability, processing integrity, or confidentiality threats from sources outside the boundaries of the system. | In addition to access rights controlled through the back-end mainframe, Rocket API provides application-layer security that can further restrict API calls by user, by function, and by data being accessed.<br><br>Data transfers are strictly between the back-end mainframe and the front-end system calling the API. There is no capability for data to be leaked, forwarded, or otherwise redirected through Rocket API.<br><br>Audit logging functionality can record all API calls, showing details of the user accessing the function, data being accessed, and data values being read and/or written. |

| Criteria | Rocket API Capabilities |
| --- | --- |
| CC5.7<br>The transmission, movement, and removal of information is restricted to authorized users and processes, and is protected during transmission, movement, or removal enabling the entity to meet its commitments and requirements as they relate to security, availability, processing integrity, or confidentiality. | All data transfers using Rocket API are secured through encrypted protocols, including TLS1.2 and SSHv2. Strong encryption provides for data security and confidentiality. |
| CC7.1<br>Security, availability, processing integrity, or confidentiality commitments and requirements, are addressed, during the system development lifecycle including design, acquisition, implementation, configuration, testing, modification, and maintenance of system components. | Any changes to the coding of an API must be processed through your source control system before being deployed through Rocket API, and are thereby subject to all of the development and change controls that you have implemented in your SDLC environment. |
| PI1.1<br>Procedures exist to prevent, detect, and correct processing errors to meet processing integrity commitments and requirements. | Transmission encryption protocols ensure the integrity of the data being transferred, to prevent technical errors or malicious interference. |
| C1.2<br>Confidential information within the boundaries of the system is protected against unauthorized access, use, and disclosure during input, processing, retention, output, and disposition in accordance with confidentiality requirements. | Rocket API leverages access credentials from the back-end mainframe operating system, and thereby inherits all access rights and restrictions associated with those credentials, including read and write capabilities.<br><br>Rocket API does not by default store any data involved with API calls, limiting the storage of such data.<br><br>Additionally, Rocket API supports data masking and anonymization capabilities to limit the exposure of data in a personally identifiable form |
| C1.3<br>Access to confidential information from outside the boundaries of the system and disclosure of confidential information is restricted to authorized parties in accordance with confidentiality commitments and requirements. | Rocket API leverages access credentials from the back-end mainframe operating system, and thereby inherits all access rights and restrictions associated with those credentials, including read and write capabilities.<br><br>In addition to access rights controlled through the back-end mainframe, Rocket API provides application-layer security that can further restrict API calls by user, by function, and by data being accessed.<br><br>Data transfers are strictly between the back-end mainframe and the front-end system calling the API. There is no capability for data to be leaked, forwarded, or otherwise redirected through Rocket API.<br><br>Customers are able to cache common API calls for performance reasons. This cached data is retained in memory only, not written to any permanent storage mechanism, and is erased when the Rocket API service is stopped, or at pre-configured time intervals. |

Rocket

rocketsoftware.com
info@rocketsoftware.com
US: 1 855 577 4323
EMEA: 0800 520 0439
APAC: 612 9412 5400
twitter.com/rocket
www.linkedin.com/company/rocket-software
www.facebook.com/RocketSoftwareInc
blog.rocketsoftware.com