

REST APIs for Password Checkout and Emergency Access

1) Password CheckOut

Syntax: **POST /rest/cmdctrl/Request**

Example - JSON payload:

```
{
  "Request": {
    "type": "PasswordCheckout",
    "runHost": "BLR-AD-SRV1_ldap",
    "reason": "Need Active Directory access",
    "duration": 360,
    "emailid": "user1@company.com"
  }
}
```

Mandatory parameters -

type: Indicates the type of operation. The value has to be "PasswordCheckout".

runHost: The application AccountDomain name to which the password checkout is requested.

reason: Enter the reason for which the access is required.

duration: Enter the duration required for the access. The value is in minutes, which ranges from 360 to 7days.

Optional parameters-

emailid: email ID of the requester.

2) Emergency Access

Syntax: **POST /rest/cmdctrl/Request**

Example - JSON payload:

```
{
  "Request": {
    "type": "EmergencyRequest",
    "accessas": 1,
    "runhost": "4.3.2.1",
    "target": "DB_ORACLE_PASSWORD",
    "reason": "Need immediate access to Oracle DB",
    "emailid": "user1@company.com",
    "requestperiod": 720,
    "force": 1
  }
}
```

Mandatory parameters -

type: Indicates the type of operation. The value has to be "EmergencyRequest".

accessas: Indicates the privileges required for accessing the system. To access the system, the values for normal user and super user are 0 and non-zero respectively.

runhost: The application AccountDomain name to which the Emergency Access is requested.

target: The type of target server. The valid values are RDP_DIRECT, RDP_SSH, pcksh, DB_ORACLE_PASSWORD, DB_ORACLE_ACCESS, DB_MSSQL_ACCESS, DB_MSSQL_PASSWORD, APPL_<subtype>.

reason: Enter the reason for which the access is required.

requestperiod: Enter the duration required for the access. The value is in minutes, which ranges from 360 to 7days.

Optional parameters -

emailid: Email ID of the requester.

3) Password CheckIn

Syntax: **PUT /rest/cmdctrl/Request**

PUT /rest/cmdctrl/Request/{id}

Example - JSON Payload:

```
{
  "Request": {
    "type": "PasswordCheckin",
    "id": "3ffc60f4-615a-4fb3-aae5-eafc3423056c",
    "comment": "Done with my work, checking in.",
    "force": 1
  }
}
```

Mandatory parameters -

type: Indicates the type of operation. The value has to be "PasswordCheckin".

id: Request ID.

Optional parameters -

force: This option is available only for the administrator. Set it to 1 to forcefully check-in a user's password checkout request.

comment: Password check-in comments.

4) Modify Emergency Access request

Syntax: **PUT /rest/cmdctrl/Request**
PUT /rest/cmdctrl/Request/{id}

4a) Example - JSON payload to 'approve' user request

Admin approves Emergency Access request

```
{
  "Request": {
    "id": "2070460c-aa83-4b5a-97c0-6141cc16a6ee",
    "type": "EmergencyRequest",
    "action": "approve",
    "comment": "You are granted access to the ldap server.",
    "runas": "-",
    "runhost": "DOM-APP_LDAP-1",
    "target": "APPL_LDAP",
    "requestperiod": 720,
    "credential": "94ced4ce-94cc-47e6-8828-8809fc3b45bd"
  }
}
```

Mandatory parameters:

id: Request ID.

action: The value has to be "approve".

type: Indicates the type of operation. The value has to be "EmergencyRequest".

requestperiod: Duration up to which the access has to be granted.

runhost: Server to which access is being granted.

runas: Account ID to whom the access is granted.

target: Type of the target server. The valid values are RDP_DIRECT, RDP, SSH, pcksh, DB_ORACLE_PASSWORD, DB_ORACLE_ACCESS, DB_MSSQL_ACCESS, DB_MSSQL_PASSWORD, APPL_<subtype>.

credential: These are the values for the following type of request:

- RDP , RDP_DIRECT & SSH - valid credential ID.
- DB_MSSQL_PASSWORD, DB_ORACLE_PASSWORD & APPL_<subtype> - "Run User@Run Host"
- DB_ORACLE_ACCESS, DB_MSSQL_ACCESS & pcksh - "null"

Optional parameters:

comment: Enter the approval comments.

4b) Example - JSON payload to 'deny' or 'revoke' user request:

Admin approves Emergency Access request

```
{
  "Request": {
    "id": "2070460c-aa83-4b5a-97c0-6141cc16a6ee",
    "type": "EmergencyRequest",
    "action": "deny",
    "comment": "You are not authorized to access."
  }
}
```

}

Mandatory parameters:

id: Request ID.

action: The value can be "deny" or "revoke".

type: Indicates the type of operation. The value has to be "EmergencyRequest".

Optional parameters:

comment: Enter the access request denial comments.

5) View the Requests

Syntax: **GET /rest/cmdctrl/Requests**

GET /rest/cmdctrl/Requests/{request_id}

Optional Parameters:

{request_id}: Get the details of the request having the ID {request_ID}. Example: "ead4248b-96b2-4022-aca9-fdd3c596952f"

All: Get the requests of all the users. Administrator only option.

queryrequest: Type of request. Valid values are 1 for emergency access and 2 for password checkout.

limit : Query result limit. Default value is 50.

status: The following are the valid status values:

1 - Emergency Access request in Pending state.

2 - Emergency Access request in Approved or Expired state.

3 - Emergency Access request that was denied by the admin.

4 - Emergency Access request that was revoked by the admin.

5 - Password checkout requests.

6 - Password checkout requests that were auto checked-in by PAM system, since the requested time had expired.

7 - Password checked in by a user.

8 - Password checked in which had failed.

delegate: Password check-in delegation states. The valid values are 0 for non-delegated, 1 for Check-in (delegated), 2 - Check-in (complete)

RequestTime : All request less than or equal to the passed value. The value is in epoch time. Example: 1448475051

6) Delete Emergency Access Request

Syntax: **DELETE /rest/cmdctrl/Request/{request_id}**

{request_id}: Request ID to be deleted. Example: "ead4248b-96b2-4022-aca9-fdd3c596952f".

Note: For more information on other REST APIs, contact the support team.