REST API Guide

Application Password CheckIn and CheckOut

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) 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.

Emergency Access

3) Emergency Access Request

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.

4) Modify Emergency Access request

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

2a) 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.

2b) 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-aca9fdd3c596952f"
 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".

Enterprise Credential Vault

7) Get Vaults Using Type

Syntax: **GET /rest/prvcrdvlt/Vaults/<type>**

Mandatory paramters:
type - Type of Account Domain. The values can be 'ssh', 'ldap', 'database', 'application', 'key', or 'sso'

Sample curl command:
curl --insecure -X GET -b
"pum_rest_auth=eJzj8ExJzSvJLKlkyOd0LC3JCMnPTs1jKGbJS8xNZU1Myc3My+TIzEtMLsksS92e4slWUlySmFvQ2r/km1Uxd1Fqbn5JanxG
fnEJQzFnQWpqEZjNmlSUmZydyBKUn5MKNKoISDFqMaDwIUYzAC0NzkzPSywpLUplqGIuzkxvZJi98ADr9wn7r2UEffVyV2rZnfZOgGN
q3Mmrt1ZfebnzgZB57mPmeNOoJtWNYhzRPRYWz3VPJXnvPdlwOKjfNPf93ZasVTWPDvRu0Kzpd+4VLFzZx+B28t3i5G8fP8g26OhdiOue
3ZKeej+sPtb9tKj2rvDQ+6vtBH6ZSl+S3fU0fFafxxf9riqmzBSRTSlpe8Pd/jZ9yZhw02rK/ZgnMuY3GBgAozN/xA==" -H "Cache-Control: no-
cache" -H "Content-Type:application/json" https://164.99.162.101/rest/prvcrdvlt/Vaults/ssh

8) Get Vaults Using ID

Syntax: **GET /rest/prvcrdvlt/Vaults/<id>**

Mandatory paramters:
id - id of the vault

Sample curl command:
curl --insecure -X GET -u admin:novell123 -H "Cache-Control: no-cache" -H "Content-Type:application/json"
https://164.99.162.101/rest/prvcrdvlt/Vault/39031fa3-b647-46f2-b0ad-76c288228a6d

9) List all the Credentials of a Vault Using the Vault ID

Syntax: **GET /rest/prvcrdvlt/Credentials/<vault_id>**

Mandatory Parameters:
vault_id - All credentials of the Vault with id, vault_id is returned

Sample curl command:
curl --insecure -X GET -b

"pum_rest_auth=eJzj8ExJzSvJLKlkyOd0LC3JCMnPTs1jKGbJS8xNZU1Myc3My+TIzEtMLsksS+3j8GQrKS5JzC1oPfbor0Exd1Fqbn5JanxGf
nEJQzFnQWpqEZgtnl1QlJiVWpyhl5RTpJeXX5aak6OXnJ+byBKUn5MKNLwISDFqMaDwIZYxAJ0RnJmel1hSWpTKUMVcnJneyKCfs0w/
YNWnk0+VX35r18ws8dDWlz+UZR6oraHjXF9Q1OnColNoPOV5dy6vYomdVaeBy5bdncqV7WbTH5z40TXfymH6zY+XGlekSnIfFMjvnrK5
d/Jms4tLeB9vLj0u8efDzSjOSfq+dVXRNyPq6hyzTXxyeu6lis8sf1GRtONn0rm1hiYTq5gyU0T0Fq7mWcHgaeDxZ6bzstvnz8hf79VhYAAau
OAxA==" -H "Cache-Control: no-cache" -H "Content-Type:application/json" https://164.99.162.101/rest/prvcrdvlt/Credentials/39031fa3-b647-
46f2-b0ad-76c288228a6d

10)   Get Credential by id

Syntax: **GET /rest/prvcrdvlt/Credential/<id>**

Mandatory parameter:
id - id of the credential

Sample curl command:
curl --insecure -X GET -b

"pum_rest_auth=eJzj8ExJzSvJLKlkyOd0LC3JCMnPTs1jKGbJS8xNZU1Myc3My+TIzEtMLsksS+3j8GQrKS5JzC1oPfbor0Exd1Fqbn5JanxGf
nEJQzFnQWpqEZgtnl1QlJiVWpyhl5RTpJeXX5aak6OXnJ+byBKUn5MKNLwISDFqMaDwIZYxAJ0RnJmel1hSWpTKUMVcnJneyKCfs0w/
YNWnk0+VX35r18ws8dDWlz+UZR6oraHjXF9Q1OnColNoPOV5dy6vYomdVaeBy5bdncqV7WbTH5z40TXfymH6zY+XGlekSnIfFMjvnrK5
d/Jms4tLeB9vLj0u8efDzSjOSfq+dVXRNyPq6hyzTXxyeu6lis8sf1GRtONn0rm1hiYTq5gyU0T0Fq7mWcHgaeDxZ6bzstvnz8hf79VhYAAau
OAxA==" -H "Cache-Control: no-cache" -H "Content-Type:application/json" https://164.99.162.101/rest/prvcrdvlt/Credential/bf40307d-284d-
4832-b905-397498b63f91

11)   Add or Modify Vault

Syntax: **PUT /rest/prvcrdvlt/Vault**
        PUT /rest/prvcrdvlt/Vault/<id>

Mandatory parameters:
type - type of vault, ssh, ldap, database, application, key, sso
profile - Applicable to ldap type domain. Value can be either 1 or 2. 1 means Active Directory, 2 means eDirectory.
Subtype - sub type of the application or shared key type vault. The vault name followed by '_<subtype>',
        example: dom-server_sap
Script - Applicable to application type only. The script that is used for password reset during password checkin process.
ConnectString - Applicable to database type vault only. The connect string to the database.
ConnectAs - Applicable to database type vault only. Value can be either 1 or 1. Connect as DB admin or as normal user respectively.

Optional parameters:
profile -
cred - id of the default credential of the Vault
id - id of the vault that is to be modified. The id could be set in the payload
        as well for modify operation. When no id is passed, it would be add operation.
host - DNS name or the IP address of the Vault. This is not applicable to key type domain.
port - port number of the service.
PasswdPolicyId - Applicable to database and application types only. Set the id of the
        password policy that is to be used for password checkin requests.
secure - Applicable to application type only. Set whether the connection to the server is
        on secure channel or not. Value can be either 0 or 1.
ScriptArgs - key/value pairs of arguments to the script of password checkin or these can be
        details about the Vault.
multiuser - Applicable to shared key type vault only. The value can be either 0 or 1.
        Determines whether the keys created within the key vault are of type multiuser or not.
        If multiuser then an integer value can be set on the keys created under it which
        determines how many can use the a key at the same time.
delegate - Applicable to database and application types. Value can be either 0 or 1. Determines
        if the password checkin should be delegated to Identity Manager.

Sample curl command: Add ssh type Vault
curl --insecure -X PUT -b

"pum_rest_auth=eJzj8ExJzSvJLKlkyOd0LC3JCMnPTs1jKGbJS8xNZU1Myc3My+TIzEtMLsksS+3j8GQrKS5JzC1oPfbor0Exd1Fqbn5JanxGf
nEJQzFnQWpqEZgtnl1QlJiVWpyhl5RTpJeXX5aak6OXnJ+byBKUn5MKNLwISDFqMaDwIZYxAJ0RnJmel1hSWpTKUMVcnJneyKCfs0w/
YNWnk0+VX35r18ws8dDWlz+UZR6oraHjXF9Q1OnColNoPOV5dy6vYomdVaeBy5bdncqV7WbTH5z40TXfymH6zY+XGlekSnIfFMjvnrK5
d/Jms4tLeB9vLj0u8efDzSjOSfq+dVXRNyPq6hyzTXxyeu6lis8sf1GRtONn0rm1hiYTq5gyU0T0Fq7mWcHgaeDxZ6bzstvnz8hf79VhYAAau

OAxA==" -H Cache-Control: no-cache -H Content-Type:application/json –d

```
'{
    "Vault": {
        "type": "ssh",
        "name": "DOM-SSH-Linux1",
        "CFG": {
            "SSH": {
                "port": "22",
                "host": "192.168.1.200"
            }
        },
        "ACL": {
            "Role": {
            }
        }
    }
}' 'https://164.99.162.101/rest/prvcrdvlt/Vault'
```

Sample curl command: Modify ssh type Vault

curl --insecure -X PUT -b

"pum_rest_auth=eJzj8ExJzSvJLKlkyOd0LC3JCMnPTs1jKGbJS8xNZU1Myc3My+TIzEtMLsksS+3j8GQrKS5JzC1oPfZ1B1sxd1Fqbn5JanxGf
nEJQzFnQWpqEZgtkV1QpFtQmptSVpyTWmxoVFxgqFthZqJrmMgSlJ+TCjS9CEgxajGg8CG2MQDdEZyZnpdYUlqUylDFXJyZ3sjAWn5+w
ppVqRf01//7Wmnb9E32iuK28+v4V1d/yJ4r3hN7fbXhMZbJr6duWPmgQTegU9+TkWuGN9uOk55rDrm82Ww9lUtr4+kippMHOhRuXhFIXGz
mdmdt4WS2U/eXrRdZ6mbr7qpSlBm19PBXtXmVbRNWzTvYcm+x4JO3unx3nLyNk+6VrJO7VsWUmSKit3A1zwoGTwOPPzOdl90+f0b+eq
8OAwMA/KuD6g==" -H Cache-Control: no-cache -H Content-Type:application/json -d '{

```
    "Vault": {
        "CFG": {
            "SSH": {
                "port": "22",
                "host": "192.168.1.201"
            }
        }
    }
}' 'https://164.99.162.101/rest/prvcrdvlt/Vault/3cfdc4a1-9d85-49d9-bdbd-bea2c4ec8c3c'
```

Sample curl command: Add application type Vault

curl --insecure -X PUT -b

"pum_rest_auth=eJzj8ExJzSvJLKlkyOd0LC3JCMnPTs1jKGbJS8xNZU1Myc3My+TIzEtMLsksS+3j8GQrKS5JzC1oPfZ1B1sxd1Fqbn5JanxGf
nEJQzFnQWpqEZgtkV1QpFtQmptSVpyTWmxoVFxgqFthZqJrmMgSlJ+TCjS9CEgxajGg8CG2MQDdEZyZnpdYUlqUylDFXJyZ3sjAWn5+w
ppVqRf01//7Wmnb9E32iuK28+v4V1d/yJ4r3hN7fbXhMZbJr6duWPmgQTegU9+TkWuGN9uOk55rDrm82Ww9lUtr4+kippMHOhRuXhFIXGz
mdmdt4WS2U/eXrRdZ6mbr7qpSlBm19PBXtXmVbRNWzTvYcm+x4JO3unx3nLyNk+6VrJO7VsWUmSKit3A1zwoGTwOPPzOdl90+f0b+eq
8OAwMA/KuD6g==" -H Cache-Control: no-cache -H Content-Type:application/json -d '{

```
    "Vault": {
        "name": "DOM-APP10_SAP",
        "type": "application",
        "CFG": {
            "PasswdPolicyId": "1",
            "Subtype": "SAP",
            "Connect": {
                "host": "192.168.1.100",
                "port": 3309,
                "secure": 1,
                "Script": "# Sample perl script for Password Reset of a user on SAP system\n\n## global variables\nmy $retVal = 1;\nmy $OS =
$^O;\nmy $cmd_output = \"\";\n\n## arguments\nmy $host = $args->arg(\"host\");\nmy $systemNumber =
$args->arg(\"systemNumber\");\nmy $clientNumber = $args->arg(\"clientNumber\");\nmy $lang = $args->arg(\"lang\");\nmy $admin =
$args->arg(\"adminName\");\nmy $adminPasswd = $args->arg(\"adminPasswd\");\nmy $user = $args->arg(\"userName\");\nmy $userPasswd =
$args->arg(\"userPasswd\");\n\n# Set passwords as environment variables\n$ENV{ADMIN_PASSWD} =
$adminPasswd;\n$ENV{USER_NEW_PASSWD} = $userPasswd;\n\n$ctx->log_info(\"*** START SAP PASSWD
RESET\");\n$ctx->log_info(\"*** Privileged Account Manager running on the OS $OS\");\n$ctx->log_debug(\"SAP System input parameters :
SAP Host - $host :: System Number - $systemNumber :: Client Number - $clientNumber :: Language :: $lang :: admin - $admin :: user - $user
\");\n$ctx->log_info(\"Resetting the password of the SAP user $user ...\");\n\n## validate inputs\nif ($host eq \"\" or $systemNumber eq \"\" or
$clientNumber eq \"\" or $admin eq \"\" or $adminPasswd eq \"\" or $user eq \"\" or $userPasswd eq \"\") {\n    $ctx->log_error(\"Incomplete
inputs - following parameters are mandatory - SAP host, systemNumber, clientNumber, admin, adminPasswd, userName and userPasswd.\");\n
return 0;\n}\n\n# set default language\nif ($lang eq \"\") {\n    $lang = \"EN\";\n}\n\n# Execute the java command for password reset\nif ($OS =~
\"^MSWin\") {\n    $cmd_output = `java -jar \"C:/\\\"Program Files\\\\\"/NetIQ/npum/service/local/cmdctrl/lib/NPUM_SAP_api.jar\" $host
$systemNumber $clientNumber $lang $admin $user`;\n} else {\n    $cmd_output = `ADMIN_PASSWD=$adminPasswd
USER_NEW_PASSWD=$userPasswd java -jar /opt/netiq/npum/service/local/cmdctrl/lib/NPUM_SAP_api.jar $host $systemNumber
$clientNumber $lang $admin $user`;\n}\n\nif ($? != 0) {\n    $ctx->log_error(\"Password reset for the user $user failed.\");\n    $retVal = 0;\n} else
{\n    $ctx->log_info(\"Succesfully resetted the password of the SAP user $user .\");\n}\n\n$ctx->log_debug(\"Command execution output as
```

below : \n $cmd_output \");\n\n$ctx->log_info(\"*** END SAP PASSWD RESET\");\nreturn $retVal;\n",
```
        "ScriptArgs": {
          "lang": "EN",
          "clientNumber": "500",
          "systemNumber": "00"
        }
      }
    }
  }
}
```
}' 'https://164.99.162.101/rest/prvcrdvlt/Vault'

Sample curl command: Add database type Vault
curl --insecure -X PUT -b
"pum_rest_auth=eJzj8ExJzSvJLKlkyOd0LC3JCMnPTs1jKGbJS8xNZU1Myc3My+TIzEtMLsksS+3j8GQrKS5JzC1oPfZ1B1sxd1Fqbn5JanxGf
nEJQzFnQWpqEZgtkV1QpFtQmptSVpyTWmxoVFxgqFthZqJrmMgSlJ+TCjS9CEgxajGg8CG2MQDdEZyZnpdYUlqUylDFXJyZ3sjAWn5+w
ppVqRf01//7Wmnb9E32iuK28+v4V1d/yJ4r3hN7fbXhMZbJr6duWPmgQTegU9+TkWuGN9uOk55rDrm82Ww9lUtr4+kippMHOhRuXhFIXGz
mdmdt4WS2U/eXrRdZ6mbr7qpSlBm19PBXtXmVbRNWzTvYcm+x4JO3unx3nLyNk+6VrJO7VsWUmSKit3A1zwoGTwOPPzOdl90+f0b+eq
8OAwMA/KuD6g==" -H Cache-Control: no-cache -H Content-Type:application/json -d '{
```
  "Vault": {
    "name": "Oracle_database",
    "type": "database",
    "CFG": {
      "PasswdPolicyId": 1,
      "ConnectString": {
        "value": "(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(Host=192.168.1.200)(Port=1521))(CONNECT_DATA=(SID=orcl)))"
      },
      "ConnectAs": {
        "value": 1
      },
      "Connect": {
        "delegate": 0
      }
    }
  }
```

}' 'https://164.99.162.101/rest/prvcrdvlt/Vault'

Sample curl command: Add shared key type Vault
curl --insecure -X PUT -b
"pum_rest_auth=eJzj8ExJzSvJLKlkyOd0LC3JCMnPTs1jKGbJS8xNZU1Myc3My+TIzEtMLsksS+3j8GQrKS5JzC1oPfZ1B1sxd1Fqbn5JanxGf
nEJQzFnQWpqEZgtkV1QpFtQmptSVpyTWmxoVFxgqFthZqJrmMgSlJ+TCjS9CEgxajGg8CG2MQDdEZyZnpdYUlqUylDFXJyZ3sjAWn5+w
ppVqRf01//7Wmnb9E32iuK28+v4V1d/yJ4r3hN7fbXhMZbJr6duWPmgQTegU9+TkWuGN9uOk55rDrm82Ww9lUtr4+kippMHOhRuXhFIXGz
mdmdt4WS2U/eXrRdZ6mbr7qpSlBm19PBXtXmVbRNWzTvYcm+x4JO3unx3nLyNk+6VrJO7VsWUmSKit3A1zwoGTwOPPzOdl90+f0b+eq
8OAwMA/KuD6g==" -H Cache-Control: no-cache -H Content-Type:application/json -d '{
```
    "Vault": {
      "name": "DOM-SSH-KEY-10",
      "type": "key",
      "CFG": {
        "Subtype": "SSH Key",
        "multiuser": 1
      },
    }
```
}' 'https://164.99.162.101/rest/prvcrdvlt/Vault'

12)   Add or Modify Credential

Syntax: **PUT /rest/prvcrdvlt/Credential**
          PUT /rest/prvcrdvlt/Credential/<id>

Mandatory parameters:
vault - id of the vault to which the credential is being added
account - name of the credential
type - 'passwd' or 'ssh_pkey', default is 'passwd'
limit - Applicable to shared keys only. If the key domain is of multiuser type,
       then the integer value can be set which determines the key usage limit.

Optional parameters:
id - id of the credential that is to be modified. The id could be set in the payload

as well for modify operation. When no id is passed, it would be add operation.
anon - applicable for LDAP credential only. Set the credential as Anonymous LDAP user. Value can be 0 or 1.
userdn - applicable for LDAP credential only. Set the DN of the credential, e.g. cn=admin,ou=users,o=myCompany

Sample curl command: Add SSH credential with password

```
curl --insecure -X PUT -b
"pum_rest_auth=eJzj8ExJzSvJLKlkyOd0LC3JCMnPTs1jKGbJS8xNZU1Myc3My+TIzEtMLsksS+3j8GQrKS5JzC1oPfZ1B1sxd1Fqbn5JanxGf
nEJQzFnQWpqEZgtkV1QpFtQmptSVpyTWmxoVFxgqFthZqJrmMgSlJ+TCjS9CEgxajGg8CG2MQDdEZyZnpdYUlqUylDFXJyZ3sjAWn5+w
ppVqRf01//7Wmnb9E32iuK28+v4V1d/yJ4r3hN7fbXhMZbJr6duWPmgQTegU9+TkWuGN9uOk55rDrm82Ww9lUtr4+kippMHOhRuXhFIXGz
mdmdt4WS2U/eXrRdZ6mbr7qpSlBm19PBXtXmVbRNWzTvYcm+x4JO3unx3nLyNk+6VrJO7VsWUmSKit3A1zwoGTwOPPzOdl90+f0b+eq
8OAwMA/KuD6g==" -H Cache-Control: no-cache -H Content-Type:application/json -d '{
    "Credential": {
        "vault" : "48784d7c-d5a1-4d0f-b8dc-13347f4a0e9f",
        "account": "root",
        "type": "pkey",
        "PCD": {
            "passwd": "abcd"
        }
    }
}' --verbose 'https://164.99.162.101/rest/prvcrdvlt/Credential'
```

Sample curl command: Add SSH credential with private key

```
curl --insecure -X PUT -b
"pum_rest_auth=eJzj8ExJzSvJLKlkyOd0LC3JCMnPTs1jKGbJS8xNZU1Myc3My+TIzEtMLsksS+3j8GQrKS5JzC1oPfZ1B1sxd1Fqbn5JanxGf
nEJQzFnQWpqEZgtkV1QpFtQmptSVpyTWmxoVFxgqFthZqJrmMgSlJ+TCjS9CEgxajGg8CG2MQDdEZyZnpdYUlqUylDFXJyZ3sjAWn5+w
ppVqRf01//7Wmnb9E32iuK28+v4V1d/yJ4r3hN7fbXhMZbJr6duWPmgQTegU9+TkWuGN9uOk55rDrm82Ww9lUtr4+kippMHOhRuXhFIXGz
mdmdt4WS2U/eXrRdZ6mbr7qpSlBm19PBXtXmVbRNWzTvYcm+x4JO3unx3nLyNk+6VrJO7VsWUmSKit3A1zwoGTwOPPzOdl90+f0b+eq
8OAwMA/KuD6g==" -H Cache-Control: no-cache -H Content-Type:application/json -d '{
    "Credential": {
        "vault": "48784d7c-d5a1-4d0f-b8dc-13347f4a0e9f",
        "account": "root2",
        "type": "ssh_pkey",
        "PCD": {
            "pkey": "-----BEGIN RSA PRIVATE KEY-----\\nProc-Type: 4,ENCRYPTED\\nDEK-Info: AES-128-
CBC,CD03D96742618AB5298EC90E12BE4A1A\\n\\nQZHu49mdFB1FFkD3VRrOL9m9SVlrIMl6C2EIdjuHS3a69prO4BYEi7Cki5LOcxgR\\
npNLYizEJgPO7NPI0VakqR3UFsxheegkFKP2TQ9AOcIM4Hiym9YZ98qraYvyVfN8h\\nJiqsj+6HVysWYx/EUGMyEw38bQYEcwSLCD6C+i
U7rAczSvORLKffYoWMZY8FO92V\\nWhoa6GW5RNgqzvTJNo3P75SARzXSVWs+0al6Wnm9cD4X66tHwQOBcaeUIa5x8Q8a\\nDxX9LX
r/1xPa+AlmLtIiwj+koCWmNKsyhl8MmQi8ihngv4/kmy7LW5w1PKj0qFyt\\nX/7Z+KH5iHWWd7exogYODqe4wfC8SudmP11rE+FFEkDjSc2
EXNwpkQ1rljP9SjjD\\nzVYNHaRkIFzp/tWPbyY2XA5WS94eSRZp0bkV0DpZFBbC1Rfk368BQH5NRCM8HIcs\\nLNemRzAmtMcFQxWH
FE8VAaxDZ26YJQbRCQ+cdeEYpns2ixUUAGc+uzcVxNhxO+sJ\\nHXssUGE0qMhLcXbXvLZ4cEpk7fp5hB22JT4+gGafR+WaWzgDfHQh8
A1FLHucKgxR\\nh7nVSzhRPi6I+DdQ5C2+FyuRpgx2+Ra+6JqOmcSLEgkN0BEhAO1HGxnXAPSiRyDv\\nz+oCIS5HbeYTJxYruWKSSp/Y
TN5cS6TuhA0YH5vWbsmV8QWf18HN4eJkgQ7gKIe4\\ntR5EF8AH2YYDRPx/NvsfX2UbMwV+TwcKoQhoRZtnzM03se05tGwBKHg7HTg
VIlQC\\nJqzfDcXb7JST+gyleL85tqCzrDA+KcIvjb2uGmRIJerouH1Df1j2kMIqCu0A9Chy\\n4IcfvBy21qi5L7BUReYcUr3b0hnQacXyrg6JS5H
JlV2AU703KNLp5AV7+wdZ53Fn\\npytDNeuTkTyK75sd1gsjOGYkcYi8r7ifaAdNHNc554/p6P9wEZDHXPCOAmp5Ws8W\\n6p5xrngOw9fN
B6FfrXCA6ukIejxJUOuPsCK/qxIrEDUH1PLLtFkA8stcW1Y1Hrgz\\nefow4UlV+lpby7T2AVFGvM29W4yJsith/39kNJWtSJgHc6h31zVd2N0
Arrbz4BNO\\npoKAhFx8ShaFTdqL3nk6CejsH80MbDY+jszuHjLdjBC4Zk+dj19NBbr/8j6XZAWw\\nr/BRcy59BqrLanynHlVSjHaaJGcqgny8c
z9CEBl2s+49kx1sBqatfh7LSBc11Kb2\\n/YcssFmzDRTCC1LLirHMZUp1KlioVsvweiVY5U5Bs/F7/RSH1nE1rdePptJpZI9L\\neyn8giirl5JvAB
dCQ6Mq7f5J/+fYkBnWv8lyOKf6wF8RuIzyet83ts4VyN9PNZJt\\nnirbTXrB57vHZZYdKFs/Z+M4C4snDyub8seYK6fCng4BtgCEXWmt8bgvd
BUqH2K6H\\nWVH0J3CojRg7qfQJuF5N8Vc7SZxsJ3OsxLEASsoSBT4doIAPKKJ4Ibv6uFyEyIGC\\n2eoL3XHdD8ikoPrFb8sBO7Lny7DbZ5
a09HwM+0D4aKz0ty7my8ArYRfmN9oix9vl\\nohqTsLXKnRL0TgkUw9rSv12xlLmxbqpOYtQQa9p+/GTOOSQBmk7+0Rev81ZYN+ti\\n-----
END RSA PRIVATE KEY-----",
            "passphrase": "mypassword"
        }
    }
}' --verbose 'https://164.99.162.101/rest/prvcrdvlt/Credential'
```

Sample curl command: Modify a credential
```
curl --insecure -X PUT -b
"pum_rest_auth=eJzj8ExJzSvJLKlkyOd0LC3JCMnPTs1jKGbJS8xNZU1Myc3My+TIzEtMLsksS+3j8GQrKS5JzC1oPfZ1B1sxd1Fqbn5JanxGf
nEJQzFnQWpqEZgtkV1QpFtQmptSVpyTWmxoVFxgqFthZqJrmMgSlJ+TCjS9CEgxajGg8CG2MQDdEZyZnpdYUlqUylDFXJyZ3sjAWn5+w
ppVqRf01//7Wmnb9E32iuK28+v4V1d/yJ4r3hN7fbXhMZbJr6duWPmgQTegU9+TkWuGN9uOk55rDrm82Ww9lUtr4+kippMHOhRuXhFIXGz
mdmdt4WS2U/eXrRdZ6mbr7qpSlBm19PBXtXmVbRNWzTvYcm+x4JO3unx3nLyNk+6VrJO7VsWUmSKit3A1zwoGTwOPPzOdl90+f0b+eq
8OAwMA/KuD6g==" -H Cache-Control: no-cache -H Content-Type:application/json -d '{
    "Credential": {
        "id" : "7fcb7cc0-688a-4eb0-a4e5-cc9f46f4431a",
        "account": "root",
```

```
      "PCD": {
          "passwd": "abcd1234"
      }
  }
}' 'https://164.99.162.101/rest/prvcrdvlt/Credential'
```

Sample curl command: Adding LDAP/Windows Credential

```
curl --insecure -X PUT -b
"pum_rest_auth=eJzj8ExJzSvJLKlkyOd0LC3JCMnPTs1jKGbJS8xNZU1Myc3My+TIzEtMLsksS+3j8GQrKS5JzC1oPfZ1B1sxd1Fqbn5JanxGf
nEJQzFnQWpqEZgtkV1QpFtQmptSVpyTWmxoVFxgqFthZqJrmMgSlJ+TCjS9CEgxajGg8CG2MQDdEZyZnpdYUlqUylDFXJyZ3sjAWn5+w
ppVqRf01//7Wmnb9E32iuK28+v4V1d/yJ4r3hN7fbXhMZbJr6duWPmgQTegU9+TkWuGN9uOk55rDrm82Ww9lUtr4+kippMHOhRuXhFIXGz
mdmdt4WS2U/eXrRdZ6mbr7qpSlBm19PBXtXmVbRNWzTvYcm+x4JO3unx3nLyNk+6VrJO7VsWUmSKit3A1zwoGTwOPPzOdl90+f0b+eq
8OAwMA/KuD6g==" -H Cache-Control: no-cache -H Content-Type:application/json -d '{
```

```
  "Credential": {
      "vault": "9386dbf0-868a-41dd-9ad7-c54a37deaf9c",
      "account": "admin",
      "type": "passwd",
      "PCD": {
          "passwd": "abcd123"
      },
      "CFG": {
          "anon": "0",
          "userdn": "cn=admin1,ou=sa,o=system"
      }
  }
}' 'https://164.99.162.101/rest/prvcrdvlt/Credential'
```

Sample curl command: Adding key in shared key domain

```
curl --insecure -X PUT -b
"pum_rest_auth=eJzj8ExJzSvJLKlkyOd0LC3JCMnPTs1jKGbJS8xNZU1Myc3My+TIzEtMLsksS+3j8GQrKS5JzC1oPfZ1B1sxd1Fqbn5JanxGf
nEJQzFnQWpqEZgtkV1QpFtQmptSVpyTWmxoVFxgqFthZqJrmMgSlJ+TCjS9CEgxajGg8CG2MQDdEZyZnpdYUlqUylDFXJyZ3sjAWn5+w
ppVqRf01//7Wmnb9E32iuK28+v4V1d/yJ4r3hN7fbXhMZbJr6duWPmgQTegU9+TkWuGN9uOk55rDrm82Ww9lUtr4+kippMHOhRuXhFIXGz
mdmdt4WS2U/eXrRdZ6mbr7qpSlBm19PBXtXmVbRNWzTvYcm+x4JO3unx3nLyNk+6VrJO7VsWUmSKit3A1zwoGTwOPPzOdl90+f0b+eq
8OAwMA/KuD6g==" -H Cache-Control: no-cache -H Content-Type:application/json -d '{
```

```
  "Credential": {
      "vault": "ed851226-efad-47e8-9444-5527e18db6fd",
      "CFG": {
          "limit": 10
      },
      "PCD": {
          "pkey": "private key",
          "passphrase": "abcd"
      },
      "type": "ssh_pkey",
      "account": "sshKey2"
  }
}' 'https://164.99.162.101/rest/prvcrdvlt/Credential'
```

13) Delete a Credential

Syntax: **DELETE /rest/prvcrdvlt/Credential/<id>**

Mandatory parameters:
id - id of the Credential object to be deleted

Sample curl command:

```
curl --insecure -X DELETE -b
"pum_rest_auth=eJzj8ExJzSvJLKlkyOd0LC3JCMnPTs1jKGbJS8xNZU1Myc3My+TIzEtMLsksS+3j8GQrKS5JzC1oPfb4kF8xd1Fqbn5JanxGf
nEJQzFnQWpqEZgtnl1QlJiVWpyhl5RTpJeXX5aak6OXnJ+byBKUn5MKNLwISDFqMaDwIZYxAJ0RnJmel1hSWpTKUMVcnJneyNBz8PW
NLRntZ681VAn9//JT0u2plM3vylrVS5MahCW6uyctzEuxYp6jq2F24fT0RLGkmeftfre4nHsmHnB+q8PHpp1y6pFxFwyD2XvebBZs+2fhVDIxs
pbFNMMjidknRcd3z+O2xs2Gu2pkN4lI+JvxC8zvX1B8zvzdyyVLN0//x7RqcemM9CqmzBQRvYWreVYweBp4/JnpvOz2+TPy13t1GBgA5++
CCQ==" -H "Cache-Control: no-cache" -H "Content-Type:application/json" https://164.99.162.101/rest/prvcrdvlt/Credential/671de945-a595-
4556-98a6-25d685c638e8
```

14) Delete a Vault

Syntax: **DELETE /rest/prvcrdvlt/Vault/<id>**

Mandatory paramters:
id - id of the Vault object to be deleted

Sample curl command:
curl --insecure -X DELETE -b
"pum_rest_auth=eJzj8ExJzSvJLKlkyOd0LC3JCMnPTs1jKGbJS8xNZU1Myc3My+TIzEtMLsksS+3j8GQrKS5JzC1oPfZ4KWsxd1Fqbn5JanxG fnEJQzFnQWpqEZgtnl1QlJiVWpyhl5RTpJeXX5aak6OXnJ+byBKUn5MKNLwISDFqMaDwIZYxAJ0RnJmel1hSWpTKUMVcnJneyJAx6wj7 C6vCk9LR5akcc27cCdFpyphnIVK47KaM6NZ/cTlLuGrjBKZtlhaRf5tsXpUm+b7hmOye8w9i53Rmu/UuP3yYqb/yvmL7ryimUp9fU/POM2+5/ cN/TsfKoscROrxpvxRb1I2mufu9ndpjLD3XpUdtW4rw/5XTq58Ht0x3x42m9mWMWUmSKit3A1zwoGTwOPPzOdl90+f0b+eq8OAwMAew yAkg==" -H "Cache-Control: no-cache" -H "Content-Type:application/json" https://164.99.162.101/rest/prvcrdvlt/Vault/39031fa3-b647-46f2-b0ad-76c288228a6

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