Saner CVEM Webservices Guide

Version 6.6

About SecPod

SecPod is a cybersecurity technology company committed to preventing cyberattacks through proactive security. Its mission is to secure computing infrastructure by enabling preventive security posture.

At the core of SecPod's offerings is the Saner Platform - a suite of solutions that help organizations establish a strong security posture to preempt cyber threats against endpoints, servers, network and cloud infrastructure, and cloud workloads. With its cutting-edge and comprehensive solutions, SecPod empowers organizations to stay ahead of evolving threats and build a resilient security framework.

303 Twin Dolphin Drive,6th Floor,
Redwood City, California 94065
USA

To learn more about SecPod, visit:
www.secpod.com

Table of Contents

Introduction

The Saner CVEM webservices guide is a comprehensive guide that provides all the needed information to integrate with Saner APIs. You can perform a wide range of operations to interact with Saner Server. Refer to the Methods Supported section for details on the supported methods in each module.

- Organization, Account, and User Management.
- Service Provisioning
- Group Management
- Download Saner Agent Installers
- Configuration Management
- Device Management
- Cyber Hygiene Score
- Posture Anomaly
- Risk Prioritization
- Compliance Management
- Patch Management
- Endpoint Management
- Reports Management
- AD Integration Management
- Network Scanner Integration Management
- Multi-factor Authentication
- Asset Management
- Vulnerability Management

Saner Server represents an Organizational hierarchy regarding accounts/sites, groups, and devices, just like how IT teams envision it in their Organization.

The following figures show a glimpse of how the Organization hierarchy is depicted in the Saner Server.

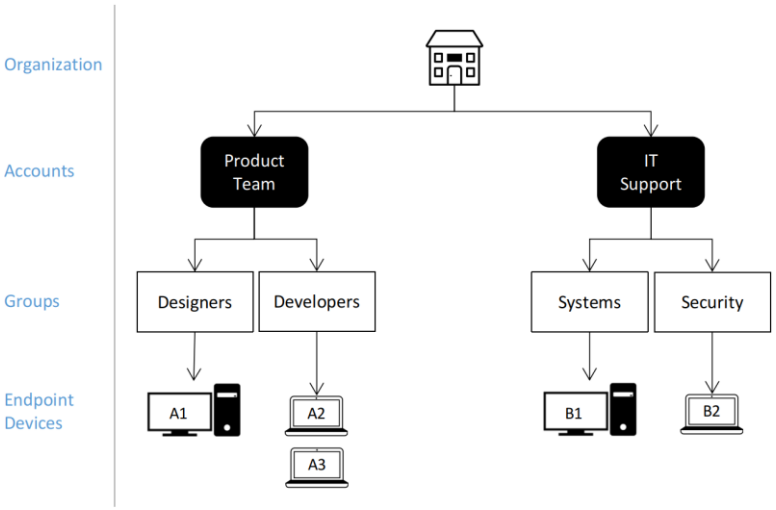


Figure 2: Organization hierarchy depicted in Saner Server

Perform – The Format of Saner Server WebServices Request

Saner Server provides a common interface to perform various operations on the server. The operations are explained in the section below:

Below is a typical request to the Saner Server:

https://saner.secpod.com/AncorWebService/perform?accountid=<account name>

'perform' is a POST request where request data is being passed as a form parameter called data. See the below Request Format for more information:

Request body in raw format:

```
{
   "request": {...}
}
```

Another parameter, *'accountid'*, corresponds to the account name managed by an administrator account. This parameter is required while performing Device, Group, Organization, Configuration, Remediation, and Reports Management. All requests require an authentication token (an API key) to be passed in the authorization parameter in the header of an HTTPS request.

Headers:

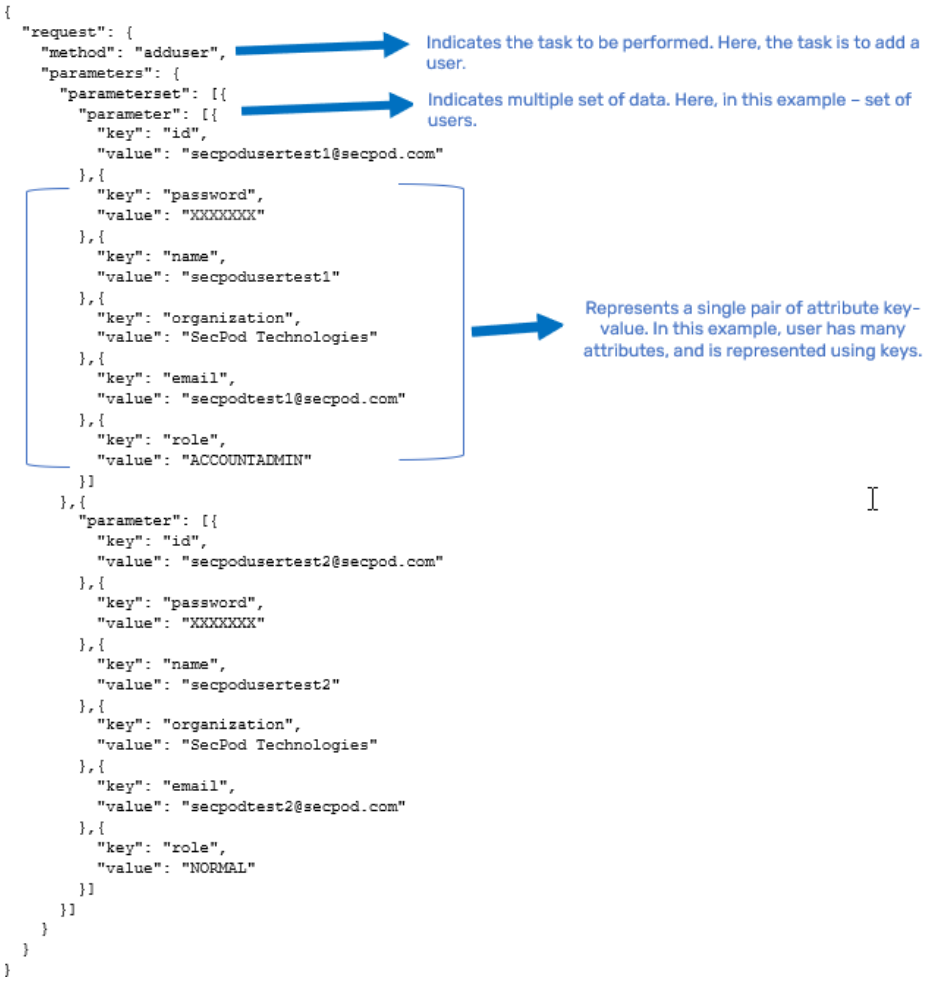
1) Content-Type: application/json

2) Authorization: <API key>

> **Note**
>
> Please get in touch with the SecPod Support team at support@secpod.com to get your API key.

Request Format

Request data is crafted in JSON Format, as shown:




```
{
  "request": {
    "method": "adduser",
    "parameters": {
      "parameterset": [{
        "parameter": [{
          "key": "id",
          "value": "secpodusertest1@secpod.com"
        },{
          "key": "password",
          "value": "XXXXXXX"
        },{
          "key": "name",
          "value": "secpodusertest1"
        },{
          "key": "organization",
          "value": "SecPod Technologies"
        },{
          "key": "email",
          "value": "secpodtest1@secpod.com"
        },{
          "key": "role",
          "value": "ACCOUNTADMIN"
        }]
      },{
        "parameter": [{
          "key": "id",
          "value": "secpodusertest2@secpod.com"
        },{
          "key": "password",
          "value": "XXXXXXX"
        },{
          "key": "name",
          "value": "secpodusertest2"
        },{
          "key": "organization",
          "value": "SecPod Technologies"
        },{
          "key": "email",
          "value": "secpodtest2@secpod.com"
        },{
          "key": "role",
          "value": "NORMAL"
        }]
      }]
    }
  }
}
```

Figure 3: Saner Server request in JSON format

Example for request json:


```
{
  "request": {
    "method": "adduser",
    "parameters": {
      "parameterset": [{
        "parameter": [{
          "key": "id",
          "value": "secpodusertest1@secpod.com"
        },{
          "key": "password",
          "value": "XXXXXXX"
        },{
          "key": "name",
          "value": "secpodusertest1"
        },{
          "key": "organization",
          "value": "SecPod Technologies"
        },{
          "key": "email",
          "value": "secpodtest1@secpod.com"
        },{
          "key": "role",
          "value": "ACCOUNTADMIN"
        }]
      },{
        "parameter": [{
          "key": "id",
          "value": "secpodusertest2@secpod.com"
        },{
          "key": "password",
          "value": "XXXXXXX"
        },{
          "key": "name",
          "value": "secpodusertest2"
        },{
          "key": "organization",
          "value": "SecPod Technologies"
        },{
          "key": "email",
          "value": "secpodtest2@secpod.com"
        },{
          "key": "role",
          "value": "NORMAL"
        }]
      }]
    }
  }
}
```

Response Format

By default, Response data is returned in the form of JSON. Responses vary from request-to-request for all get queries (for example: get account details, remediation details, device details, etc). However, for all other requests, a common response is received. The common Response format is shown below:

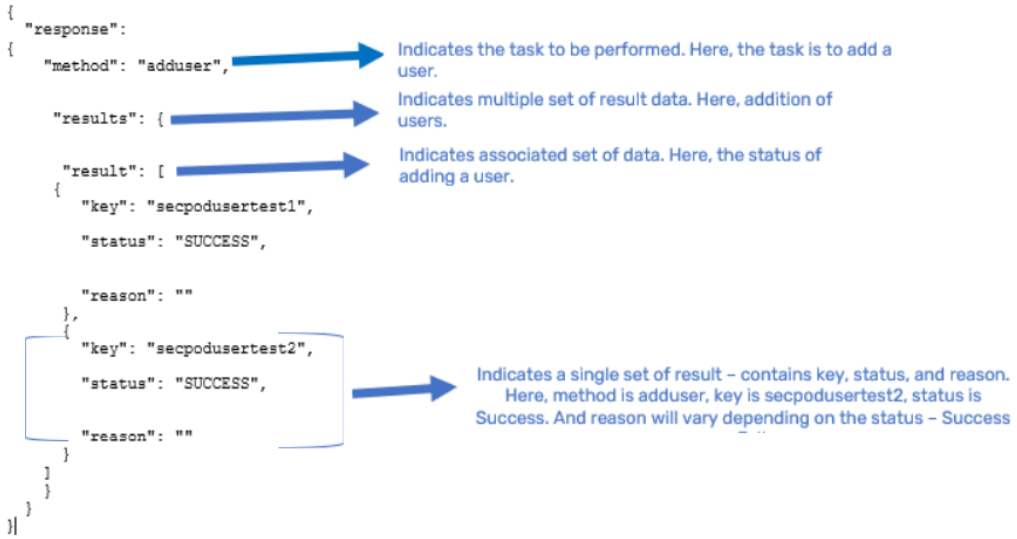


Figure 4: Saner Server response in JSON format

Example response JSON in case of success

```
{
  "response": {
    "method": "adduser",
    "results": {
      "result": [{
        "key": "secpodusertest1",
        "status": "SUCCESS",
        "reason": ""
      },{
        "key": "secpodusertest2",
        "status": "SUCCESS",
        "reason": ""
      }]
    }
  }
}
```

Example response JSON in case of failure:

```
{
  "response": {
    "method": "adduser",
    "results": {
      "result": [{
        "key": "secpodusertest1",
        "status": "FAIL",
        "reason": "Failed due to duplicate name"
      },{
        "key": "secpodusertest2",
        "status": "SUCCESS",
        "reason": ""
      }]
    }
  }
}
```

For few of the APIs, the response will be a compressed file (ZIP format). Zip file will contain response.json and response.sig files.

List of APIs returning response in ZIP format are as follows:

- downloAdadAgent
- downloadAgent
- getAgentActivationConf
- getDevicePdfReport
- getPdfReport

Request Schema


```
{
  "schema": "http: //json-schema.org/draft-04/schema#",
  "id": "http: //jsonschema.net",
  "type": "object",
  "properties": {
    "request": {
      "id": "http: //jsonschema.net/request",
      "type": "object",
      "properties": {
        "method": {
          "id": "http: //jsonschema.net/request/method",
          "type": "string"
        },
        "parameters": {
          "id": "http: //jsonschema.net/request/parameters",
          "type": "array",
          "items": [{
            "id": "http: //jsonschema.net/request/parameters/0",
            "type": "object",
            "properties": {
              "parameterset": {
                "id": "http: //jsonschema.net/request/parameters/0/parameterset",
                "type": "object",
                "properties": {
                  "parameter": {
                    "id": "http:
//jsonschema.net/request/parameters/0/parameterset/parameter",
                    "type": "string"
                  },
                  "value": {
                    "id": "http: //jsonschema.net/request/parameters/0/parameterset/value",
                    "type": "string"
                  }
                }
              }
            }
          }]
        }
      },
      "required": [
        "method",
        "parameters"
      ]
    }
  },
  "required": [
    "request"
  ]
}
```

Response Schema


```
{
  "$schema": "http://json-schema.org/draft-04/schema#",
  "id": "http://jsonschema.net",
  "type": "object",
  "properties": {
    "response": {
      "id": "http://jsonschema.net/response",
      "type": "object",
      "properties": {
        "method": {
          "id": "http://jsonschema.net/response/method",
          "type": "string"
        },
        "results": {
          "id": "http://jsonschema.net/response/results",
          "type": "object",
          "properties": {
            "result": {
              "id": "http://jsonschema.net/response/results/result",
              "type": "array",
              "items": {
                "id": "http://jsonschema.net/response/results/result/0",
                "type": "object",
                "properties": {
                  "key": {
                    "id": "http://jsonschema.net/response/results/result/0/key",
                    "type": "string"
                  },
                  "status": {
                    "id": "http://jsonschema.net/response/results/result/0/status",
                    "type": "string"
                  },
                  "reason": {
                    "id": "http://jsonschema.net/response/results/result/0/reason",
                    "type": "string"
                  }
                }
              }
            }
          }
        }
      },
      "required": [
        "method",
        "results"
      ]
    }
  },
  "required": [
    "response"
  ]
}
```

Methods Supported

Organization Management

- addOrganization
- updateOrganization
- removeOrganization
- getOrganization
- assignUserToOrganization
- unassignUserFromOrganization

Account and User Management

- addAccount
- removeAccount
- updateAccount
- getAccount
- addUser
- removeUser
- updateUser
- getUser
- assignUserAccount
- unassignUserAccount
- updatePassword
- getAdmin
- getAuditActionCodes
- getAuditlogs

Service Provisioning

- updateServiceProvision
- getServiceProvision

Group Management

- addGroup
- updateGroup
- deleteGroup
- getGroup
- assignDevicestoGroup
- unassignDevicesFromGroup

Download Saner Agent Installers

- downloadAgent
- getDownloadUrl
- getAgentActivationConf

Configuration Management

- addConfig
- getConfig
- updateConfig
- deleteConfig

Device Management

- addDevice
- updateDevice
- deleteDevice
- movedevice
- pinandupindevice
- getDevice
- getDeviceDetails
- getDeviceVulnerabilities
- getDeviceAccountInfo
- scanDevice
- getDeviceReport
- getBasicSystemDetails
- getDeviceTagKeys
- getInstalledApplications
- getDeviceJobInfo
- getHostResults
- getDeviceJobSummary
- getDeviceJobDetails
- updateDeviceAlias
- removeDeviceAlias
- uninstallAgent

Cyber Hygiene Score

- getAccountHygieneScore
- calculateHygieneScore
- getCHScoreForDeviceWithStatus
- getCHScoreSummaryForFamily
- getCHScoreSummaryForOS
- getCHScoreSummaryForGroup
- getTrendingCHScore
- getCHSFrequencyDistribution
- getTopCHSAttributesofPA
- getTopCHSAttributesofVM
- getTopCHSAttributesofCM
- getTopCHSAttributesofPM
- getOrgCHScoreByFamily
- getOrgCHScoreByGroup
- getOrgCHScoreByOs
- getOrgHygieneScore
- getTopCHSAttributes
- getCHSWeightage
- updateCHSWeightage
- getTrendingOrgHygieneScore
- getCHScanStatus
- updateOrgCHSWeightage

Asset Exposure

- addWhiteListedAssets
- addBlackListedAssets

Posture Anomaly

- initiatePostureAnomalyScan
- getPostureAnomalyScannerConfig

- addPostureAnomalyScannerConfig
- deletePostureAnomalyScannerConfig
- getStatus
- getPostureAnomaly
- getConfiguration
- getConfigurationStatus
- getWhitelist
- postWhitelist
- getAllConfiguration

Vulnerability Management

- excludevulnerability

Compliance Management

- addBenchmark
- deleteBenchmark
- getAllBenchmarkProvisions
- provisionBenchmark
- updateProvisionBenchmark
- deleteProvisionBenchmark
- getApplicableMisconfigurationRemediation
- createMisconfigurationRemediationJob
- getNonSecJobDetails
- getMisconfigurationFixforRollback
- createMisconfigurationRollbackTask
- getMisconfigurationRollbackStatus
- deleteMisconfigurationRollbackTask
- getAutomationRuleStatus
- updateAutomationRuleStatus

Risk Prioritization

- getMissionCriticalDeviceData
- saveMissionCriticalDeviceData
- getRiskPrioritizationSummary
- getRiskSummary
- getRiskDetails
- getRiskMitigationDetails
- getCriticalAssetRiskDetails
- getRiskAutomatabilityDetails
- getRiskTechnicalImpactDetails
- getRiskExploitabilityDetails
- getDeviceRiskDetails
- getRPJson
- getRiskPrioritizationStatus
- getRiskonMissionCritical
- getRiskonMissionPrevalence
- getRiskonEssentialDevices
- prioritizeRisks
- getRPTrends
- getChainableRisks
- getMVEDetails

Patch Management

- getApplicableRemediation
- createRemediationJob
- getRemediationJobStatus
- getRemediationJob
- getRemediationJobPatch
- deleteRemediation
- getApplicableNonSecurityRemediation
- getApplicableFirmwareRemediation
- createFirmwareRemediationJob
- getFirmwareRemediationJobStatus
- deleteFirmwareRemediationJob
- getAllApplicableRules
- getRemediationRule
- addRemediationRule
- updateRemediationRule
- deleteRemediationRule
- getAutomationRuleStatus
- updateAutomationRuleStatus
- getPatchesforRollback
- createPatchRollbackTask
- getPatchRollbackStatus
- deletePatchRollbackTask
- rebootDevice
- getRebootTaskStatus
- deleteReboottask
- excludevulnerability

Endpoint Management

- addSoftwareDeployment
- addSoftwareProvision
- uploadSoftwareUrl
- uploadInstallerPackage
- uploadCompressedFile
- uninstallSoftware
- getAllApplications

Reports Management

- getProfileReport
- getVulnerableAssets
- getAssets
- getAssetsByVulnerability
- getProfileNames
- getReportNames
- getReportApis
- getReportApiData
- getPdfReport
- getDevicePdfReport
- addReportProvision
- deleteReportProvision

AD Integration Management

- addADconfig
- updateADconfig
- deleteADconfig
- getADconfig
- initiateADscan
- downloadADagent
- testADconnection
- getADconnectionStatus
- addADscanExcludeList
- removeADscanExcludeList
- getADscanExcludeList
- getADscanMergedData
- applyADscanChanges
- getADscanActionStatus

Network Scanner Integration Management

- addNetworkScanner
- removeNetworkScanner
- getNetworkScanStatus
- getDiscoveryScanStatus
- initiateNetworkScan
- initiateDiscoveryScan
- addDiscoveryScanConfig
- removeDiscoveryScanConfig
- removeNetworkScanConfig
- addNetworkScanConfig
- isNetworkScanner
- isNetworkScanConfigAssigned
- assignNetworkScanConfig
- unassignNetworkScanConfig
- getNetworkScanPolicy
- isNetworkScanPolicyAssigned
- getAllNetworkScanPolicyNames
- getAllNetworkScanConfigNames
- stopDiscoveryScan
- stopNetworkScan
- getNetworkScanConfig
- updateNetworkScanConfig
- getDiscoveryScanConfig
- updateDiscoveryScanConfig

Multi-Factor Authentication

- addMFAPolicy
- updateMFAPolicy
- removeMFAPolicy
- getMFAPolicy
- isMFAPolicyExist
- updateUserMFAPolicy
- getUserMFAPolicy
- isUserMFAPolicyExist
- enforceMultiFactor
- withdrawMultiFactor
- isMultiFactorEnforced

- isMultiFactorEnabled

Organization, Account, and User Management

Saner server provides various APIs for managing organizations, accounts, and user management tasks. You can add, remove, update, retrieve organizations/accounts/users, and assign/unassign users to organizations/accounts. This section provides detailed insights into the Organization, Account, and User Management APIs and their usage.

Add Organization

This method allows you to add organizations in Saner Server. An organization needs a unique name, email address, end (expiry) date, and number of subscriptions. The value will be set to the administrator's end date if you do not provide one. Please ensure that the value end date does not exceed the value of your administrator's end date.

Method Name: addOrganization

Method Type: POST

Mandatory Parameters: name, email, numberofsubscriptions, and enddate

Sample request	Sample response
<pre>{ "request": { "method": "addOrganization", "parameters": { "parameterset": [{ "parameter": [{ "key": "name", "value": "testorganization1" },{ "key": "email", "value": "secpodtest@secpod.com" },{ "key": "numberofsubscriptions", "value": "10" },{ "key": "enddate", "value": "2024-10-10" }] }] } } }</pre>	In case of success: <pre>{ "response": { "method": "addOrganization", "results": { "result": [{ "key": "testorganization1", "status": "success", "reason": "" }] } } }</pre>

Possible Error Cases

- Organization not created.
- Organization name must be of minimum 4 characters.
- Invalid organization name. Only alphanumeric characters and -, _, . are allowed.
- Failed due to invalid email.
- Failed due to invalid subscription count (expected integer and should not exceed maximum available subscription count)
- Failed due to invalid expiry date (expected format yyyy-mm-dd and should not exceed your license validity)
- No records found with given name.
- Field <key> cannot be empty.
- Subscription value should be greater than zero.

Update Organization

This method allows you to update an organization's information in Saner Server. It requires the name of the organization to be updated and all other details.

Method Name: updateOrganization

Method Type: POST

Mandatory Parameters: name

Sample request

```
{
  "request": {
    "method": "updateOrganization",
    "parameters": {
      "parameterset": [{
        "parameter": [{
          "key": "name",
          "value": "testorganization1"
        },{
          "key": "email",
          "value": "user@test.com"
        },{
          "key": "numberofsubscriptions",
          "value": "5"
        },{
          "key": "enddate",
          "value": "2024-10-10"
        }]
      }]
    }
  }
}
```

To update organization name itself use 'newname' attribute, example:

```
{
  "request": {
    "method": "updateOrganization",
    "parameters": {
      "parameterset": [{
        "parameter": [{
          "key": "name",
          "value": "testorganization1"
        },{
          "key": "newname",
          "value": "Newtestorganization1"
        }]
      }]
    }
  }
}
```

Sample response

In case of success:

```
{
  "response": {
    "method": "updateOrganization",
    "results": {
      "result": [{
        "key": "testorganization1",
        "status": "success",
        "reason": ""
      }]
    }
  }
}
```

Possible Error Cases

- Organization not updated.
- Organization name must be of minimum 4 characters.
- Invalid organization name. Only alphanumeric characters and -, _ , . are allowed.
- Failed due to invalid email.
- Failed due to invalid subscription count (expected integer and should not exceed maximum available subscription count)
- Failed due to invalid expiry date (expected format yyyy-mm-dd and should not exceed your license validity)
- No records found with given name.

- Field <key> cannot be empty.
- Subscription value should be greater than zero.

Remove Organization

This method allows you to remove organizations in Saner Server. It only requires the name of organizations to be deleted.

Method Name: removeOrganization

Method Type: POST

Mandatory Parameters: name

Sample request	Sample response
{ "request": { "method": "removeOrganization", "parameters": { "parameterset": [{ "parameter": [{ "key": "name", "value": "testorganization1" }] }] } } }	In case of success: { "response": { "method": "removeOrganization", "results": { "result": [{ "key": "testorganization1", "status": "success", "reason": "" }] } } }

Possible Error Cases

- Organization not removed.
- Invalid organization name. Only alphanumeric characters and -, _, . are allowed.
- No records found with given name.
- Field <key> cannot be empty.

Get Organization

This method allows you to get details of organizations in the Saner Server. It only requires the name of the organization to get its information. All organization details get fetched if the organization name is not provided.

Method Name: getOrganization

Method Type: POST

Mandatory Parameters: organization

Sample request	Sample response
{ "request": { "method": "getOrganization", "parameters": { "parameterset": [{	In case of success: { "organizations": [{ "organizationinfo": { "name": "testorganization1",

<table>
<tr>
<td>

```
        "parameter": [{
          "key": "organization",
          "value": "testorganization1"
        }]
      }]
    }
  }
}
```

</td>
<td rowspan="2">

```
      "email": "user@test.com",
      "startdate": "2023-09-12",
      "enddate": "2024-10-10",
      "maxsubscriptions": 5,
      "inusesubscriptions": 0
    }
  }]
}
```

</td>
</tr>
<tr>
<td>

To get all organization details:

```
{
  "request": {
    "method": "getOrganization"
  }
}
```

</td>
</tr>
</table>

Possible Error Cases

- Invalid organization name. Only alphanumeric characters and -, _, . are allowed.
- Field <key> cannot be empty.

Assign User to Organization

This method allows you to assign an ORGADMIN user to an organization in Saner Server. It requires user id and organization name.

Method Name: assignUserToOrganization

Method Type: POST

Mandatory Parameters: user and organization

Sample request	Sample response
`{ "request": { "method": "assignUserToOrganization", "parameters": { "parameterset": [{ "parameter": [{ "key": "user", "value": "user@test.com" },{ "key": "organization", "value": "testorganization1" }] }] } } }`	In case of success: `{ "response": { "method": "assignUserToOrganization", "results": { "result": [{ "key": "user@test.com", "status": "success", "reason": "" }] } } }`

Possible Error Cases

- Invalid organization name. Only alphanumeric characters and -, _,. are allowed.
- User Update Failed

Unassign User from Organization

This method allows you to unassign an ORGADMIN user from an organization in the Saner Server. It requires a user id and organization name.

Method Name: unassignUserFromOrganization

Method Type: POST

Mandatory Parameters: user and organization

Sample request	Sample response
{ "request": { "method": "unassignUserFromOrganization", "parameters": { "parameterset": [{ "parameter": [{ "key": "user", "value": ""user@test.com"" },{ "key": "organization", "value": "testorganization1" }] }] } } }	In case of success: { "response": { "method": "unassignUserFromOrganization", "results": { "result": [{ "key": ""user@test.com"", "status": "success", "reason": "" }] } } }

Possible Error Cases

- Invalid organization name. Only alphanumeric characters and -, _, are allowed.
- User Update Failed

Add Account

This method allows you to add an account in Saner Server. An account requires a unique account name, organization name, email address, start date, expiry date, and number of subscriptions. In case start date and/or end date is not provided, values will default to the administrator's default values. Please ensure that the start and end date values do not precede or exceed your account's start date or end date values(respectively). The attribute agent auto-upgrade represents that if agents under the account are automatically upgraded, the default value is true. The field organization name must be an existing organization in the Saner server.

Method Name: addAccount

Method Type: POST

Mandatory Parameters: name, organization, email, and numberofsubscriptions

Sample request	Sample response
{ "request": {	In case of success: {

<table>
<tr>
<td>

```
    "method": "addaccount",
    "parameters": {
      "parameterset": [{
        "parameter": [{
          "key": "name",
          "value": "testaccount"
        },{
          "key": "organization",
          "value": "ABC Corp"
        },{
          "key": "email",
          "value":"user@test.com"
        },{
          "key": "numberofsubscriptions",
          "value": "100"
        },{
          "key": "agentautoupgrade",
          "value": "false"
        }]
      }]
    }
  }
}
```

</td>
<td>

```
  "response": {
    "method": "addaccount",
    "results": {
      "result": [{
        "key": "testaccount",
        "status": "SUCCESS",
        "reason": ""
      }]
    }
  }
}
```

In case of failure:

```
{
  "response": {
    "method": "addaccount",
    "results": {
      "result": [{
        "key": "testaccount",
        "status": "FAIL",
        "reason": "Failed due to duplicate
name"
      }]
    }
  }
}
```

</td>
</tr>
</table>

Possible Error Cases

- Failed due to duplicate name.
- Failed due to invalid subscription count (expected integer and should not exceed maximum available subscription count)
- Failed due to invalid expiry date (expected format yyyy-mm-dd and should not exceed your license validity)
- Failed due to invalid start date (expected format yyyy-mm-dd and should not precede or exceed your license validity)
- Failed due to invalid email.
- Failed due to invalid account name or ID.
- Field <key> cannot be empty.
- Account Creation Failed (in all other cases) Invalid account type. Value can be cloud, endpoint, or workload.
- Invalid account type.

Remove Account

This method allows you to remove one or more accounts in Saner Server. It only requires the name of accounts to be deleted.

Method Name: removeAccount

Method Type: POST

Mandatory Parameters: name

<table>
<tr><th>Sample request</th><th>Sample response</th></tr>
<tr>
<td>

```
{
  "request": {
    "method": "removeaccount",
    "parameters": {
      "parameterset": [{
        "parameter": [{
          "key": "name",
```

</td>
<td>

```
{
  "response": {
    "method": "removeaccount",
    "results": {
      "result": [{
        "key": "Test Account",
        "status": "SUCCESS",
```

</td>
</tr>
</table>

<table>
<tr>
<td>

```
          "value": "Test Account"
        }]
      },{
        "parameter": [{
          "key": "name",
          "value": "Test Account 1"
        }]
      }]
    }
  }
}
```

</td>
<td>

```
        "reason": ""
      },{
        "key": "Test Account 1",
        "status": "FAIL",
        "reason": "No records found with given
name"
      }]
    }
  }
}
```

</td>
</tr>
</table>

Possible Error Cases

- Failed due to no records found with given name.
- Account Deletion Failed (in all other cases)

Update Account

This method allows you to update details of one or more accounts in Saner Server. It requires the name of accounts to be updated and all other details. Organization added via add account api previously will be fetched automatically and used and need not be explicitly mentioned in the request.

Method Name: updateAccount

Method Type: POST

Mandatory Parameters: name and email

<table>
<tr><th>Sample request</th><th>Sample response</th></tr>
<tr>
<td>

```
{
 "request": {
  "method": "updateaccount",
  "parameters": {
   "parameterset": [
    {
     "parameter": [
      {
       "key": "name",
       "value": "Test Account"
      },{
       "key": "email",
       "value": "testaccount@secpod.com"
      }
     ]
    },{
     "parameter": [
      {
       "key": "name",
       "value": "Test Account 1"
      },
      {
       "key": "email",
       "value":
"testaccount1@secpod.com"
      }]}]}}}
```

</td>
<td rowspan="2">

```
{
  "response": {
    "method": "updateaccount",
    "results": {
      "result": [{
        "key": "Test Account",
        "status": "SUCCESS",
        "reason": ""
      },{
        "key": "Test Account 1",
        "status": "FAIL",
        "reason": "No records found with the
given name"
      }]
    }
  }
}
```

</td>
</tr>
<tr>
<td>

To update the account name itself, use the ‘newname’ attribute:

```
{
  "request": {
    "method": "updateaccount",
```

</td>
</tr>
</table>

Sample request	Sample response
<pre>"parameters": { "parameterset": [{ "parameter": [{ "key": "name", "value": "Test Account 1" },{ "key": "newname", "value": "newaccounttest1" }] }] } } }</pre>	

Possible Error Cases

- Failed due to no records found with given name.
- Failed due to invalid subscription count (expected integer and should not exceed maximum available subscription count)
- Failed due to invalid expiry date (expected format yyyy-mm-dd and should not exceed your license validity)
- Failed due to invalid start date (expected format yyyy-mm-dd and should not precede or exceed your license validity.
- Failed due to invalid email.
- Account Update Failed (in all other cases)
- Invalid account type. Value can be cloud, endpoint,or workload

Get Account Details

This method allows you to get details of one or multiple accounts in Saner Server. It only requires the name of accounts to be updated. If no name is provided, all account details are fetched. The attribute '*agentautoupgrade'* represents if agents under the account is automatically upgraded, default value is true.

Method Name: getAccount

Method Type: POST

Mandatory Parameters: name

Sample request	Sample response
<pre>{ "request": { "method": "getaccount" } }</pre>	<pre>{ "accountinfo": { "account": [{ "name": "secpodaccounttest1", "organization": "SecPod Technologies", "email": "secpodtest1@secpod.com", "startdate": "2023-01-01", "enddate": "2024-12-31", "maxsubscriptions": "100", "inusesubscriptions": "10", "agentautoupgrade": false },{ "name": "secpodaccounttest2", "organization": "SecPod Technologies", "email": "secpodtest2@secpod.com", "startdate": "2023-01-01", "enddate": "2024-12-31", "maxsubscriptions": "100", "inusesubscriptions": "0", "agentautoupgrade": true }] }</pre>
Request to get specific accounts: <pre>{ "request": { "method": "getaccount", "parameters": { "parameterset": [{ "parameter": [{ "key": "name", "value": "secpodaccounttest1" }] },{ "parameter": [{ "key": "name", "value": "secpodaccounttest2" }] }]</pre>	

<pre> } } }</pre>	<pre>}</pre>

Possible Error Cases

- Failed due to no records found with given name.
- Login ID should be provided (if organization is given).
- Organization should be provided (if login id is given).
- Failed due to invalid account name or id.
- Failed due to invalid organization.
- Failed due to invalid login id.

Add User

This method allows you to add one or more users in Saner Server. Each user requires a unique login id, an email ID, name, password, a combination of upper-case and lower-case alphabets, numbers and at least one special character, organization name, email address, start date, expiry date, and role. If start date and/or end date are not provided, values will default to administrators' default values.

Ensure that your password meets the below requirements.

1. Password should have at least 8 characters.
2. Password should not exceed 100 characters.
3. Password should contain at least one digit.
4. Password should have lowercase characters.
5. Password should have uppercase characters.
6. Password should have at least one non-alphanumeric character.

Please ensure that the start and end date values do not precede or exceed your account's start date or end date values(respectively).

The role of a user is either NORMAL – this user will have access to one or more accounts or ACCOUNTADMIN - which is an Account administrator who can manage multiple normal users and their access to accounts or ORGADMIN – which is an organization administrator who can manage multiple users and their access to accounts. It is also required to pass a parameter named '*maxsubscriptions*' which indicates the total number of subscriptions given to that administrator. There is an additional role called ADMIN. This admin user can manage all the organizations and accounts.

Following is the hierarchy of User roles:

- ADMIN
- ORGADMIN
- ACCOUNTADMIN
- NORMAL

Method Name: **addUser**

Method Type: POST

Mandatory Parameters: name, password, organization, email, and role.

Sample request	Sample response
<pre>To create ACCOUNTADMIN and NORMAL users: { "request": { "method": "adduser", "parameters": {</pre>	<pre>In case of successful creation of users: { "response": { "method": "adduser", "results": {</pre>

Sample request	Sample response
<pre>"parameterset": [{ "parameter": [{ "key": "id", "value": "secpodusertest1@secpod.com" },{ "key": "password", "value": "XXXXXXX" },{ "key": "name", "value": "secpodusertest1" },{ "key": "UserGroup", "value": "IT Admin" },{ "key": "email", "value": "secpodtest1@secpod.com" },{ "key": "role", "value": "ACCOUNTADMIN" }] },{ "parameter": [{ "key": "id", "value": "secpodusertest2@secpod.com" },{ "key": "password", "value": "XXXXXXX" },{ "key": "name", "value": "secpodusertest2" },{ "key": "UserGroup", "value": "SecPod Technologies" },{ "key": "email", "value": "secpodtest2@secpod.com" },{ "key": "role", "value": "NORMAL" }] }] } } }</pre>	<pre>"result": [{ "key": "secpodusertest1", "status": "SUCCESS", "reason": "" },{ "key": "secpodusertest2", "status": "SUCCESS", "reason": "" }] } } }</pre> In case of failure: <pre>{ "response": { "method": "adduser", "results": { "result": [{ "key": "secpodusertest1", "status": "FAIL", "reason": "Failed due to duplicate name" },{ "key": "secpodusertest2", "status": "SUCCESS", "reason": "" }] } } }</pre>

Possible Error Cases

- Failed due to duplicate name.
- Failed due to invalid administrator ID.
- Failed due to invalid email.
- User Creation Failed (in all other cases).

Remove User

This method allows you to remove one or more users in Saner Server. It only requires ids of users to be deleted.

Method Name: removeUser

Method Type: POST

Mandatory Parameters: id

Sample request	Sample response
<pre>{ "request": { "method": "removeuser", "parameters": { "parameterset": [{ "parameter": [{ "key": "id", "value": "secpodusertest1@secpod.com" }] },{ "parameter": [{ "key": "id", "value": "secpodusertest3@secpod.com" }] }] } } }</pre>	<pre>{ "response": { "method": "removeuser", "results": { "result": [{ "key": "secpodusertest1@secpod.com ", "status": "SUCCESS", "reason": "" },{ "key": "secpodusertest3@secpod.com ", "status": "FAIL", "reason": "No records found with given name" }] } } }</pre>

Possible Error Cases

- Failed due to no records found with given name.
- User Deletion Failed (in all other cases).

Update User

This method lets you update the details of one or more users in the Saner Server. It requires the IDs of users and all other information to be updated. The key ‘role’ accepts the following values: ACCOUNTADMIN, ORGADMIN, and NORMAL.

Method Name: updateUser

Method Type: POST

Mandatory Parameters: id

Sample request	Sample response
<pre>{ "request": { "method": "updateUser", "parameters": { "parameterset": [{ "parameter": [{ "key": "id", "value": "user1@secpod.com" }, { "key": "newid", "value": "user2@secpod.com" },</pre>	<pre>{ "response": { "method": "updateuser", "results": { "result": [{ "key": "user1@secpod.com ", "status": "SUCCESS", "reason": "" },{ "key": "user2@secpod.com ", "status": "FAIL", "reason": "No records found with given name" }]</pre>

Sample request	Sample response
<pre>{ "key": "name", "value": "testuser" }, { "key": "UserGroup", "value": "secpod" }, { "key": "role", "value": "NORMAL" }] }] } } }</pre>	<pre>} } }</pre>

Possible Error Cases

- Failed due to no records found with given name.
- Failed due to invalid administrator ID.
- Failed due to invalid email.
- User Update Failed (in all other cases).
- Failed due to duplicate login ID.

Get User Details

This method allows you to get details of one or more users in Saner Server. It only requires login ID of users. If no name is provided, all users' details are fetched.

Method Name: getUser

Method Type: POST

Mandatory Parameters: id

Sample request	Sample response
<pre>{ "request": { "method": "getuser" } }</pre>	<pre>{ "users": [{ "userinfo": { "id": "john_doe@secpod.com", "role": "ACCOUNTADMIN", "name": "Account Admin", "organization": "", "manageableOrganizationAndAccounts": [{ "organization": "Documentation", "accounts": ["Demo Account"] }], "UserGroup": "OrgAdmins", "email": "john_doe@secpod.com",</pre>

Sample request	Sample response
	"startdate": "2024-10-08", "enddate": "2025-12-15", "twofactorauth": false, "multifactorauth": false, "adminid": "admin@secpod.com" } }, { "userinfo": { "id": "test_user@gmail.com", "role": "NORMAL", "name": "Test User", "organization": "", "manageableOrganizationAndAccounts": [{ "organization": "Documentation", "accounts": ["Integration Demo", "Feature Demo", "Demo Account"] }], "UserGroup": "Demo", "email": "test_user@gmail.com", "startdate": "2024-10-10", "enddate": "2025-12-15", "twofactorauth": false, "multifactorauth": false, "adminid": "admin@secpod.com" } }] }
{ "request": { "method": "getuser", "parameters": { "parameterset": [{ "parameter": [{ "key": "id", "value": "new_user@secpod.com" }] }, { "parameter": [{ "key": "id", "value": "patch_admin@secpod.com" }] }] } } }	{ "users": [{ "userinfo": { "id": "new_user@secpod.com", "role": "NORMAL", "name": "Read_Only_User", "organization": "", "manageableOrganizationAndAccounts": [{ "organization": "Documentation", "accounts": ["Integration Demo", "Feature Demo", "Demo Account"] }], "UserGroup": "test", "email": "new_user@secpod.com", "startdate": "2025-04-03", "enddate": "2025-12-15", "twofactorauth": false, "multifactorauth": false, "adminid": "admin@secpod.com" } }, { "userinfo": { "id": "patch_admin@secpod.com", "role": "ACCOUNTADMIN", "name": "Patch Admin",

Sample request	Sample response
	"organization": "", "manageableOrganizationAndAccounts": [{ "organization": "Documentation", "accounts": ["Integration Demo", "Feature Demo", "Demo Account"] }], "UserGroup": "Account Administrator", "email": "patch_admin@secpod.com", "startdate": "2025-04-03", "enddate": "2025-12-15", "twofactorauth": false, "multifactorauth": false, "adminid": "admin@secpod.com" } }] }

Possible Error Cases

- Failed due to no records found with given name.

Assign User to Account

This method allows you to assign a NORMAL or ACCOUNTADMIN user to an account in Saner Server. Users will be assigned to respective organizations based on their account. Organization admin cannot be assigned to an account, if tried, error response is returned. It requires user Id and account name.

Method Name: assignUserAccount

Method Type: POST

Mandatory Parameters: user and account

Sample request	Sample response
{ "request": { "method": "assignuseraccount", "parameters": { "parameterset": [{ "parameter": [{ "key": "user", "value": "secpodusertest1@secpod.com" },{ "key": "account", "value": "secpodaccounttest1" }] },{ "parameter": [{ "key": "user", "value": "secpodusertest3@secpod.com"	{ "response": { "method": "assignuseraccount", "results": { "result": [{ "key": "secpodusertest1@secpod.com", "status": "SUCCESS", "reason": "" },{ "key": "secpodusertest3@secpod.com", "status": "FAIL", "reason": "User Update Failed" }] } } }

<pre>},{ "key": "account", "value": "secpodaccounttest2" }] }] } } }</pre>	

Possible Error Cases

- User Update Failed.

Unassign User to Account

This method allows you to unassign a '*NORMAL'* or '*ACCOUNTADMIN'* user to an account in Saner Server. The user will be unassigned from the respective organization if none of the accounts under that particular organization is assigned to user. Organization admin cannot be unassigned from an account, if tried, error response is returned. It requires user id and account name.

Method Name: unassignUserAccount

Method Type: POST

Mandatory Parameters: user and account

Sample request	Sample response
<pre>{ "request": { "method": "unassignuseraccount", "parameters": { "parameterset": [{ "parameter": [{ "key": "user", "value": "secpodusertest1@secpod.com" },{ "key": "account", "value": "secpodaccounttest1" }] },{ "parameter": [{ "key": "user", "value": "secpodusertest3@secpod.com" },{ "key": "account", "value": "secpodaccounttest2" }] }] } } }</pre>	<pre>{ "response": { "method": "unassignuseraccount", "results": { "result": [{ "key": "secpodusertest1@secpod.com", "status": "SUCCESS", "reason": "" },{ "key": "secpodusertest3@secpod.com", "status": "FAIL", "reason": "User update failed" }] } } }</pre>

Possible Error Cases

- User Update Failed

Update User Password

This method allows you to update the password of a user in Saner Server. It requires user id and password.

Method Name: updatePassword

Method Type: POST

Mandatory Parameters: id and password

Sample request	Sample response
<pre>{ "request": { "method": "updatepassword", "parameters": { "parameterset": [{ "parameter": [{ "key": "id", "value": "secpodusertest1@secpod.com" },{ "key": "password", "value": "P4ssw0rd@123" }] },{ "parameter": [{ "key": "id", "value": "secpodusertest3@secpod.com" },{ "key": "password", "value": "P4ssw0rd@123" }] }] } } }</pre>	<pre>{ "response": { "method": "updatepassword", "results": { "result": [{ "key": "secpodusertest1@secpod.com ", "status": "SUCCESS", "reason": "" },{ "key": "secpodusertest3@secpod.com ", "status": "FAIL", "reason": "User update failed” }] } } }</pre>

Possible Error Cases

- User Update Failed.

Get Admin Details

This method allows you to fetch admin details such as subscription details, number of accounts managed by the admin.

Method Name: getAdmin

Method Type: POST

Mandatory Parameters: id

Sample request	Sample response
<pre>{ "request": { "method": "getadmin", "parameters": { "parameterset": [{ "parameter": [{</pre>	<pre>{ "users": [{ "userinfo": { "id": "testuser@secpod.com", "role": "ADMIN", "name": "secpod",</pre>

```
          "key": "id",
          "value": "testuser@secpod.com"
        }]
      }]
    }
  }
}
```

```
      "organization": "SecPod Technologies",
      "email": "testuser@secpod.com",
      "startdate": "2023-06-17",
      "enddate": "2024-12-29",
      "twofactorauth": false,
      "adminid": "",
      "maxsubscriptions": 3000,
      "inusesubscriptions": 1458,
      "accounts": 56
    }
  }]
}
```


Possible Error Cases

- Failed due to invalid login id.
- Field <key> cannot be empty.

Get Audit Action Codes

This method fetches the action codes and values of the audit logs.

Method Name: getAuditActionCodes

Method Type: POST

Mandatory Parameters: tool

Sample request


```
{
 "request": {
  "method": "getauditactioncodes",
  "parameters": {
   "parameterset": [
    {"parameter": [
      {"key": "tool",
       "values": [
        "8000",
        "5000"
       ]}]}]}}}
```


Sample response


```
{
 "auditcodes": [
  {
   "code": 8000,
   "name": "EndPoint Management",
   "values": [
    {
     "code": 8001,
     "value": "Query Creation"
    },
    {
     "code": 8002,
     "value": "Query removal"
    },
    {
     "code": 8003,
     "value": "Query update"
    },]}}
```


Possible Error Cases

- Invalid action code.
- Invalid tool code was given.
- No tool found.

Get Audit Logs

This method is used to fetch audit logs based on organization, accounts, users, startdate and enddate, tool, actions, and limit.

Method Name: getAuditLogs

Method Type: POST

Mandatory Parameters: startdate and enddate

Sample request

```
{
 "request": {
  "method": "getauditlogs",
  "parameters": {
   "parameterset": [
    {
     "parameter": [
      {
       "key": "organization",
       "value": "Test_Org"
      },
      {
       "key": "accounts",
       "values": [
        "Test_Account1",
        "Test_Account2"
       ]
      },
      {
       "key": "users",
       "values": [
        "user1@secpod.com"
       ]
      },
      {
       "key": "startdate",
       "value": "2023-01-31"
      },
      {
       "key": "enddate",
       "value": "2024-02-01"
      },
      {
       "key": "limit",
       "value": "200"
      }
     ]
    }
   ]
  }
 }
}
```

Sample response

```
{
 "auditlogs": [
  {
   "timestamp": "2023-02-01 11:57:56 AM
UTC",
   "user_UID": "user1@secpod.com",
   "org_UID": "Test_Org",
   "account_UID": "Test_Account2",
   "tool_code": 3000,
   "tool_message": "Account Management",
   "action_code": 3001,
   "action_message": "Account Access",
   "message": "Account access Test_Account2"
  },
  {
   "timestamp": "2023-02-01 05:26:04 AM
UTC",
   "user_UID": "admin@secpod.com",
   "org_UID": "",
   "account_UID": "Test_Account2",
   "tool_code": 14000,
   "tool_message": "Network Scanner
Management",
   "action_code": 14019,
   "action_message": "Updated Device As
Network Scanner",
   "message": "Network Scanner Upgrade
Issued For Host sp-test-laptop"
  }
 ]
}
```

Possible Error Cases

- Invalid action code.
- Invalid limit.
- Incorrect Date format.
- Provide date in yyyy-MM-dd in UTC format.

Service Provisioning

Saner Server provides interfaces for service provisioning, allowing you to add, remove, update, and get services. The following services are available within Saner:

1. Asset Exposure (AE)
2. Posture Anomaly (PA)
3. Vulnerability Management (VM)
4. Compliance Management (CM)
5. Risk Prioritization (RP)
6. Patch Management (PM)
7. Endpoint Management (EM)

This section provides insights into Saner CVEM Service Provisioning APIs and their usage.

Add/Remove/Update Service Provision

This method allows you to add, remove or update a service provision in Saner Server.

Value 1 stands for Enable, 0 for Disable.

Value -1 can be provided to remove the service provision. Use 1 to add, 0 or -1 to remove a service.

Value 0 can be used only for account service provision

Mandatory Values are: Service name, Account name for account service provision, Organization name for Organization service provision.

Method Name: updateServiceProvision

Method Type: POST

Mandatory Parameters: service, account

Sample request	Sample response
<pre>{ "request": { "method": "updateserviceprovision", "parameters": { "parameterset": [{ "parameter": [{ "key": "service", "value": "vulnerabilitymanagement" },{ "key": "enabled", "value": 0 },{ "key": "account", "value": "secpodaccounttest1" }] },{ "parameter": [{ "key": "service", "value": "endpointmanagement" },{ "key": "enabled", "value": 0</pre>	In case of success: <pre>{ "response": { "method": "updateserviceprovision", "results": { "result": [{ "key": "secpodaccounttest1", "status": "SUCCESS", "reason": "" }] } } }</pre>
	In case of failure: <pre>{ "response": { "method": "updateserviceprovision", "results": { "result": [{ "key": "secpodaccounttest1", "status": "FAIL",</pre>

Sample request	Sample response
<pre>},{ "key": "account", "value": "secpodaccounttest1" }] },{ "parameter": [{ "key": "service", "value": "compliancemanagement" },{ "key": "enabled", "value": 0 },{ "key": "account", "value": "secpodaccounttest1" }] },{ "parameter": [{ "key": "service", "value": "assetmanagement" },{ "key": "enabled", "value": 1 },{ "key": "account", "value": "secpodaccounttest1" }] },{ "parameter": [{ "key": "service", "value": "patchmanagement" },{ "key": "enabled", "value": 1 },{ "key": "account", "value": "secpodaccounttest1" }] },{ "parameter": [{ "key": "service", "value": "postureanomaly" },{ "key": "enabled", "value": 1 },{ "key": "account", "value": "secpodaccounttest1" }] }] } } }</pre>	<pre>"reason": "Failed to update account service" }] } } }</pre>

Possible Error Cases

- Field <key> cannot be empty.
- Failed due to invalid account name or Id.
- Failed to update service.
- Failed due to invalid role.
- Failed to update account service.
- Failed due to invalid login id.
- Failed due to invalid login id.
- Failed due to invalid organization.
- Failed to update organization service.
- Failed to update user service.
- Organization does not exist.
- Failed due to invalid service name.

- Failed due to invalid enable value.
- Provide either user or account or organization.
- Field service cannot be empty.
- Enable value 0 is not valid for user <user>.
- Enable value 0 is not valid for organization <organization>.
- Provision restricted for <service> service.

Get Service Provision

This method allows you to get service provision details of admin user, account or organization in Saner Server.

Value 1 stands for Enabled, 0 for Disabled and -1 stands for Not Provisioned.

Method Name: getServiceProvision

Method Type: POST

Mandatory Parameters: No mandatory parameters needed.

Sample request	Sample response
<pre>{ "request": { "method": "getServiceProvisionDetails", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "Test Account" }]}]}}}</pre>	<pre>In case of success: { "serviceprovision": { "account": { "endpoint": { "assetmanagement": 1, "auditing": -1, "compliancemanagement": 1, "devicemanagement": 1, "edr": 1, "endpointmanagement": 1, "patchmanagement": 1, "reportsalertsmanagement": 1, "riskprioritization": 1, "vulnerabilitymanagement": 1 } }, "user": { "endpoint": { "assetmanagement": 1, "auditing": -1, "compliancemanagement": 1, "devicemanagement": 1, "edr": 1, "endpointmanagement": 1, "patchmanagement": 1, "reportsalertsmanagement": 1,</pre>

Sample request	Sample response
	<pre>"riskprioritization": 1, "vulnerabilitymanagement": 1 } } } }</pre>

Possible Error Cases

- Unable to fetch account details.

Group Management

You can create groups and assign devices to them in Saner CVEM. You can perform operations such as adding, updating, and deleting a group. This section provides insights into the Saner CVEM Group Management APIs and their usage.

Add Group

This method allows you to add a group. Creation of group requires unique group name and description. These fields should only contain alphanumeric characters, space, dot(.), underscore(_) or hyphen(-). Max allowed characters is 50. 'accountid' should be passed as part of the following query parameter: https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: addGroup

Method Type: POST

Mandatory Parameters: groupname and groupdesc

Sample request

```
{
  "request": {
    "method": "addgroup",
    "parameters": {
      "parameterset": [{
        "parameter": [{
          "key": "groupname",
          "value": "my-group"
        },{
          "key": "groupdesc",
          "value": "group for xyz devices"
        }]
      }]
    }
  }
}
```

Sample response

In case of success:

```
{
  "response": {
    "method": "addgroup",
    "results": {
      "result": [{
        "key": "my-group",
        "status": "SUCCESS",
        "reason": ""
      }]
    }
  }
}
```

In case of failure:

```
{
  "response": {
    "method": "addgroup",
    "results": {
      "result": [{
        "key": "my-group",
        "status": "FAIL",
        "reason": "Group name exists"
      }]
    }
  }
}
```

Possible Error Cases

- Invalid Input.
- Group name exists.
- Invalid Group name.
- Invalid Group description.
- Unknown exception occurred.

Update Group

This method allows you to update one or more groups. It requires groupname to be updated and allows you to update description or assign profile and configuration settings to the group. It is required to provide complete details of group (description, profile name, configuration name etc) each time this update request is sent. If configuration name or profile name is not provided during update, it is understood that configuration or profile is to be unassigned from that group. 'accountid' should be passed as part of the following query parameter: https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: updateGroup

Method Type: POST

Mandatory Parameter: groupname

Sample request	Sample response
<pre>{ "request": { "method": "updategroup", "parameters": { "parameterset": [{ "parameter": [{ "key": "groupname", "value": "my-group" },{ "key": "newgroupname", "value": "my-newgroup" },{ "key": "groupdesc", "value": "group for Microsoft Windows 8 devices" },{ "key": "profilename", "value": "profile1" },{ "key": "configname", "value": "myconfig" }] }] } } }</pre>	In case of success: <pre>{ "response": { "method": "updategroup", "results": { "result": [{ "key": "my-newgroup", "status": "SUCCESS", "reason": "" }] } } }</pre>
	In case of failure: <pre>{ "response": { "method": "updategroup", "results": { "result": [{ "key": "my-group", "status": "FAIL", "reason": "Invalid Group name" }] } } }</pre>

Possible Error Cases

- Invalid Input.
- Invalid Group name.
- Invalid new group name.
- Invalid Group description.
- Unknown exception occurred.
- No operation to perform.
- No Records.
- Account not found.

Delete Group

This method allows you to delete one or multiple groups. It requires group names to be deleted. All devices if assigned to that group will be unassigned from the group upon deletion. 'accountid' should be passed as part of the following query parameter:
https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: deleteGroup

Method Type: POST

Mandatory Parameters: groupname

Sample request	Sample response
<pre>{ "request": { "method": "deletegroup", "parameters": { "parameterset": [{ "parameter": [{ "key": "groupname", "value": "my-group" }] },{ "parameter": [{ "key": "groupname", "value": "my-group1" }] }] } } }</pre>	In case of success: <pre>{ "response": { "method": "deletegroup", "results": { "result": [{ "key": "my-group", "status": "SUCCESS", "reason": "" }] } } }</pre>
	In case of failure: <pre>{ "response": { "method": "deletegroup", "results": { "result":[{ "key":"my-group", "status":"FAIL", "reason":"Invalid Group name" }] } } }</pre>

Possible Error Cases

- Invalid Input.
- Invalid Group name.
- Invalid Group description.
- Unknown exception occurred.
- No operation to perform.
- No Records.

Get Group

This method allows you to get one or more groups' information. It requires 'groupnames' to be fetched. 'accountid' should be passed as part of the following query parameter:
https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: getGroup

Method Type: POST

Mandatory Parameters: groupname

Sample request	Sample response
<pre>{ "request": { "method": "getgroup", "parameters": { "parameterset": [{ "parameter": [{ "key": "groupname", "value": "windows 7" }] }, { "parameter": [{ "key": "groupname", "value": "ubuntu" }] }] } } }</pre>	<pre>In case of success: { "groups": { "group": [{ "name": "windows 7", "description": "Devices which have windows 7 installed", "confId": "", "profileId": "WINDOWS_7_ISO_27001_COMPLIANCE (1431756874409)", "devices": { "device": [{ "name": "acer-pc" },{ "name": "abc-pc" },{ "name": "def-pc " }] } },{ "name": "ubuntu", "description": "Devices which have ubuntu installed", "confId": "", "profileId": "", "devices": { "device": [{ "name": "xyz-pc" }] } }] } }</pre>

Possible Error Cases

- Invalid Input.
- No Records.

Assign Devices To Group

This method is an interface to assign devices to their corresponding groups. Group names and hostnames are the required fields. One device can only be associated with a single group. 'accountid' should be passed as part of the following query parameter:
https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: assignDevicesToGroup

Method Type: POST

Mandatory Parameters: groupname and hostname

Sample request	Sample response
<pre>{ "request": { "method": "assigndevicestogroup", "parameters": { "parameterset": [{ "parameter": [{ "key": "groupname", "value": "my-group" },{ "key": "hostname", "value": "myhost2" },{ "key": "hostname", "value": "myhost1" }] }] } } }</pre>	<pre>In case of success: { "response": { "method": "assigndevicestogroup", "results": { "result": [{ "key": "myhost1", "status": "SUCCESS", "reason": "" },{ "key": "myhost2", "status": "SUCESS", "reason": "" }] } } }</pre>

Possible Error Cases

- Invalid Input.
- Invalid Group name.
- “Unknown exception occurred” would be returned if devices were already assigned to a group.
- Host does not belong to the given group.
- Group cannot be updated for pinned device(s).
- Field key cannot be empty
- Invalid host name

Unassign Devices From Group

This method is an interface to unassign devices from their corresponding groups. If hostname is empty for a group, it will remove all devices associated with that group. ‘accountid’ should be passed as part of the following query parameter: https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: unassignDevicesFromGroup

Method Type: POST

Mandatory Parameters: groupname and hostname

Sample request	Sample response
<pre>{ "request": { "method": "unassigndevicesfromgroup", "parameters": { "parameterset": [{ "parameter": [{ "key": "groupname", "value": "my-group" },{ "key": "hostname", "value": "myhost2"</pre>	<pre>{ "response": { "method": "unassigndevicesfromgroup", "results": { "result": [{ "key": "myhost1", "status": "SUCCESS", "reason": "" },{ "key": "myhost2", "status": "SUCCESS",</pre>

<table>
<tr>
<td>

```
        },{
          "key": "hostname",
          "value": "myhost1"
        }]
      }]
    }
  }
}
```

</td>
<td>

```
        "reason": ""
      }]
    }
  }
}
```

</td>
</tr>
</table>

Possible Error Cases

- Invalid Input.
- Invalid Group name.
- Host does not belong to the given group.
- Group cannot be updated for pinned device(s).
- Invalid host name.
- Field key cannot be empty.
- Host already belongs to the given group.

Download Saner Agent Installers

Saner Agent can be installed and configured using the APIs. You can use these APIs to download, configure, and activate Saner Agents.This section provides detailed information about all the Saner Agent Installer APIs and their usage.

Download Agent

This method allows you to download multiple Saner Agents based on an account. The required fields are account name, type and architecture. The key ‘type’ can have the following values: osx, exe, rpm, dpkg, apk or all and ‘architecture’ can have the following values: x86, x64 or all. The value exe is for downloading Microsoft Windows installers, rpm is for RPM based machines, dpkg is for downloading Debian machines, osx is for Mac OS X machines and apk is for downloading Debian machines.

Method Name: downloadAgent

Method Type: POST

Mandatory Parameters: accountid, type, and architecture.

<table>
<tr><th>Sample request</th><th>Sample response</th></tr>
<tr>
<td rowspan="2">

```
{
  "request": {
    "method": "downloadagent",
    "parameters": {
      "parameterset": [{
        "parameter": [{
          "key": "accountid",
          "value": "test"
        },{
           "key": "type",
          "value": "dpkg"
        },{
          "key": "architecture",
```

</td>
<td>

```
In case of success, you would get a compressed file (ZIP format) with agent installers.
```

</td>
</tr>
<tr>
<td>

```
In case of failure:

{
  "response": {
    "method":"downloadagent",
    "results": {
      "result":[{
        "key":"null",
        "status":"FAIL",
        "reason":"Could not retrieve agents for download"
```

</td>
</tr>
</table>

<table>
<tr>
<td>

```
          "value": "x86"
        }]
      }]
    }
  }
}
```

</td>
<td>

```
      }]
    }
  }
}
```

</td>
</tr>
</table>

Possible Error Cases

- Failed due to unable to retrieve agents for download.
- Provide a valid agent type.
- Provide a valid agent architecture.
- Provide a supported agent architecture.
- Field key cannot be empty.

Get Agent Installer URL

This method allows you to get the URL to download multiple Saner Agents based on an account.

Method Name: getDownloadUrl

Method Type: POST

Mandatory Parameters: accountname

<table>
<tr><th>Sample request</th><th>Sample response</th></tr>
<tr>
<td rowspan="2">

```
{
  "request": {
    "method": "getdownloadurl",
    "parameters": {
      "parameterset": [{
        "parameter": [{
          "key": "accountname",
          "value":
"secpodaccounttest1"
        }
      }]
    }
  }
}
```

</td>
<td>

In case of success:

```
{
  "accountinfo": {
    "account": [{
      "name": "secpodaccounttest1",
      "url": "https://saner.secpod.com/download…"
    },{
      "name": "secpodaccounttest2",
      "url": "https://saner.secpod.com/download…"
    }]
  }
}
```

</td>
</tr>
<tr>
<td>

In case of failure:

```
{
  "response": {
    "method":"getdownloadurl",
    "results": {
      "result":[{
        "key":"secpodaccounttest3",
        "status":"FAIL",
        "reason":"Could not retrieve agents for
download"
      }]
    }
  }
}
```

</td>
</tr>
</table>

Possible Error Cases

- Failed due to unable to retrieve Saner Agents for download.

.

Get Agent Activation Conf

This method allows you to download activation conf file for an account. Activation conf file is mandatory for the agent installation. Get Agent activation conf requires account name to be provided. Activation conf file will be unique per account and same conf file can be used with multiple build types (Windows, Linux or Mac) within the account.

Method Name: getAgentActivationConf

Method Type: POST

Mandatory Parameters: accountid

Sample request	Sample response
<pre>{ "request": { "method": "getagentactivationconf", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "Test" }] }] } } }</pre>	<pre>In case of success, you would get a compressed file (ZIP format) with agent activation conf file. In case of failure: { "response": { "method": "getagentactivationconf", "results": { "result": [{ "key": "", "status": "FAIL", "reason": "Invalid Account" }] } } }</pre>

Possible Error Cases

- Invalid Account
- Expired Admin
- Expired Account

Configuration Management

Saner Server provides interfaces to add, update, and delete Saner Agent configurations. This section provides insights into the Saner Agent Configuration Management APIs and their usage.

To specify scan settings

1. **RunMode:** Choose one of the following:
    a. Full Throttle — This scan setting ensures a speedy scan.
    b. Low — This scan mode works well if you have limited system resources.
2. **ScanTypes:** Mention the following scan type with comma separation.
    a. VULNERABILITY – enables scan for vulnerabilities.
    b. COMPLIANCE – enables scan for misconfiguration.
3. **ScheduledScanTime:** The time at which scan starts. It is denoted as HH:MM:AM or HH:MM:PM. For example, 12:00:PM
4. **ScheduleDownloadTime:** The time at which content download will start. It is denoted as HH:MM:AM or HH:MM:PM. For example, 11:00:AM.

To specify remediation settings

1. **RemediationJobPollInterval:** The time interval in minutes to configure agents to poll for settings change, remediation jobs or rules set by administrator. Example: a value 5 shows that agents would poll every 5 minutes for any changes set by administrator.
2. **VendorProductPatchServer:** Always set to default. This specifies agents to retrieve the updates from the Update server configured for your machine for Vendor Patches such as Microsoft.
3. **ThirdPartyProductPatchServerType:** Can be set to
    a. default - This specifies agents to contact the SecPod server for any updates to third party products such as Adobe Reader, VLC Player.
    b. local - Allows administrators or users to specify a server in their organization that serves remediation content to machines in their network. Setting up a local machine and synchronizing it at regular intervals avoids multiple downloads over the Internet.
4. When local is selected, **LocalResourcesURL** must be specified. The local server could be an HTTP/HTTPS server or any FTP server. Example: http://192.168.1.196 or ftp://192.168.1.196. Supported protocols are HTTP, HTTPS, and FTP.
5. **VendorProductInstallationType:** Vendor updates installation type could be:

a. Quiet – Agents would silently install patches without any user intervention. It is suggested that quiet mode should be selected for seamless patching and configuration during auto-remediation mode.
b. Interactive – Agents would prompt an installation screen on endpoints and would require user intervention.

6. **ThirdPartyProductInstallationType:** Vendor updates installation type could be:
   a. Quiet – Agents would silently install patches without any user intervention. It is suggested that quiet mode should be selected for seamless patching and configuration during auto-remediation mode.
   b. Interactive – Agents would prompt an installation screen on the endpoints and would require user intervention.
7. **BufferPatchesEnabled**, set to true or false. This feature when enabled, buffers patches from the server to be used for remediation, thereby reducing remediation time.
8. **BufferPatchesDirectorySize**, Maximum cache size limit (in MB) to buffer patches. By default, it is 1024.
9. **BufferSpeedInPercent**, maximum percentage of bandwidth to be consumed while buffering patches. For example, 70.

To specify network proxy settings

1. **ProxyEnabled**, set to true to enable proxy settings, otherwise false.
2. **ProxyURL**, URL of proxy server
3. **ProxyPort**, Port of proxy server
4. **ProxyAuthEnabled**, set to true if proxy server has authentication settings, otherwise false.
5. **ProxyUsername**, username for authentication of proxy server.
6. **ProxyPassword**, password for authentication of proxy server.

To specify upgrade options

1. **Saner AutoUpgrade, configuration for agents to upgrade.**
   a. **Enabled - upgrade themselves automatically.**
   b. Manual - using manual intervention.
   c. Disabled - disable any further upgrade.
2. **PreferredLanguage**, set to EN. Currently only English (EN) is supported.
3. **hostpeerverification,** is used to verify if agent is contacting valid server. By default it will be “true” for cloud and it can’t be changed, for in-house servers it will be “false” by default and user can update it.
4. **auditlog,** set to “on” or “onwithdebug”, based on this agent will decide log format.
5. **cputhreshold,** field will decide what percentage of CPU the agent process can use in the system.
6. **missedscanaction,** field will decide if a scan is missed what step should agent take.
7. **secondaryPatchServer**, is remediation patch backup server set by user.
8. **secondaryPatchServerWhen,** field will tell when should agent contact secondary patch server.

Note

Only one set of configuration settings can be applied to a group. To create configuration settings, you can initiate method 'addconfig' and to update existing settings, use 'updateconfig' method.

You can also remove or get details of configuration settings using 'deleteconfig' and 'getconfig' respectively.

Add Configuration Settings

This method allows you to add Agent configuration settings. Creation of settings requires unique names and description. These fields should only contain alphanumeric characters, space, dot(.), underscore(_) or hyphen(-). 'accountid' should be passed as part of the following query parameter: https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: addConfig

Method Type: POST

Mandatory Parameters: configname and configdesc

Sample request

```
{
  "request": {
    "method": "addconfig",
    "parameters": {
      "parameterset": [{
        "parameter": [{
          "key": "configname",
          "value": "myconfig"
        },{
          "key": "configdesc",
          "value": "config-for-yxz-group"
        }]
      }]
    }
  }
}
```

Sample response

In case of success:

```
{
  "response": {
    "method": "addconfig",
    "results": {
      "result": [{
        "key": "myconfig",
        "status": "SUCCESS",
        "reason": ""
      }]
    }
  }
}
```

In case of failure:

```
{
  "response": {
    "method": "addconfig",
    "results": {
      "result": [{
        "key": "myconfig",
        "status": "FAIL",
        "reason": "Configuration name exists"
      }]
    }}}
```

Possible Error Cases

- Invalid Input
- Account not found
- Configuration name exists.
- Invalid configuration name
- Improper configuration description
- Unknown exception occurred.

Get Configuration Setting Details

This method allows you to get one or more of the Saner Agent configuration setting information. 'accountid' should be passed as part of the following query parameter:
https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: getConfig

Method Type: POST

Mandatory Parameters: no mandatory parameters needed.

Sample request	Sample response
<pre>{ "request": { "method": "getconfig", "parameters": { "parameterset": [{ "parameter": [{ "key": "configname", "value": "myconfig" }] }] } } }</pre>	In case of success: <pre>{ "settings": [{ "configuration": { "description": "this is default config", "name": "myconfig", "configuration-settings": { "configuration-setting": [{ "key": "RunMode", "value": "Full throttle" },{ "key": "ScanTypes", "value": "VULNERABILITY,COMPLIANCE" },{ "key": "ScheduledScanTime", "value": "12:00:PM" },{ "key": "ScheduledDownloadTime", "value": "11:00:AM" },{ "key": "AutoRemediation", "value": "OFF" },{ "key": "VendorProductPatchServer", "value": "default" },{ "key": "ThirdPartyProductPatchServerType", "value": "default" },{ "key": "LocalResourcesURL", "value": "HTTP://" },{ "key": "VendorProductInstallationType", "value": "quiet" },{ "key": "ThirdPartyProductInstallationType", "value": "quiet" },{ "key": "RemediationJobPollInterval", "value": "5" },{ "key": "BufferPatchesEnabled", "value": "false" },{ "key": "BufferPatchesDirectorySize", "value": "1024"</pre>

Sample request	Sample response
	`},{ "key": "BufferSpeedInPercent", "value": "70" },{ "key": "PreferredLanguage", "value": "EN" },{ "key": "ProxyEnabled", "value": "false" },{ "key": "ProxyURL", "value": "" },{ "key": "ProxyPort", "value": "" },{ "key": "ProxyAuthEnabled", "value": "false" },{ "key": "ProxyUsername", "value": "" },{ "key": "ProxyPassword", "value": "" },{ "key": "AgentAutoUpgrade", "value": "manual" },{ "key": "HostPeerVerification", "value": "true" }] } } }]}`

Possible Error Cases

- Invalid Input.
- No Records.
- Field Config Name cannot be empty.

Update Existing Configuration Settings

This method allows you to update one or more Agent configuration settings. It requires configuration setting name (key: configname) to be updated and allows you to update description (key: configdesc) and other setting key values such as 'runmode', 'scantype', 'scheduledscantime', 'scheduledownloadtime', 'autoremediation', 'remediationjobpollinterval', 'vendorproductpatchserver', 'thirdpartyproductpatchservertype', 'proxyurl', 'vendorproductinstallationtype', 'thirdpartyproductinstallationtype', 'bufferpatchesenabled', 'proxyport', 'bufferpatchesdirectorysize', 'bufferspeedinpercent', 'proxyusername', 'proxypassword', 'agentautoupgrade', 'hostpeerverification'. 'accountid' should be passed as part of the following query parameter: https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

The input for "hostpeerverification" must be true for cloud server.

Method Name: updateConfig

Method Type: POST

Mandatory Parameters: configname

Sample request	Sample response
{ "request": { "method": "updateconfig", "parameters": { "parameterset": [{ "parameter": [{ "key": "configname", "value": "myconfig" },{ "key": "runmode", "value": "Low|Full Throttle" },{ "key": "scantype", "value": "VULNERABILITY" },{ "key": "scheduledscantime", "value": "11:00:PM" },{ "key": "scheduledownloadtime", "value": "10:00:PM" },{ "key": "remediationjobpollinterval", "value": "2" },{ "key": "autoremediation", "value": "off" },{ "key": "vendorproductpatchserver", "value": "Default|default,windowsupdate" },{ "key": "thirdpartyproductpatchservertype", "value": "default|Local" },{ "key": "localresourcesurl", "value": "http|https|ftp://url" },{ "key": "vendorproductinstallationtype", "value": "quiet" },{ "key": "thirdpartyproductinstallationtype", "value": "quiet" },{ "key": "secondarypatchserverwhen", "value": "Default Server Connect fails|Default Server Patch Search fails" },{ "key": "bufferpatchesenabled", "value": "false" },{ "key": "proxyurl", "value": "https://192.168.XX.XX" },{ "key": "proxyport", "value": "8080" },{ "key": "proxyusername", "value": "user-001" },{ "key": "proxypassword", "value": "secpod"	In case of success: { "response": { "method": "updateconfig", "results": { "result": [{ "key": "myconfig", "status": "SUCCESS", "reason": "" }] } } } In case of failure: { "response": { "method": "updateconfig", "results": { "result": [{ "key": "myconfig", "status": "FAIL", "reason": "Invalid configuration name" }] } } }

Sample request	Sample response
<pre>},{ "key": "agentautoupgrade", "value": "disabled" },{ "key": "auditlog", "value": "on|onwithdebug" },{ "key": "missedscanaction", "value": "As soon as it detects missed scan|Postpone by x hours|nextscheduledtime" },{ "key": "AgentInteractionMode", "value": "Live|Poll" },{ "key": "HostPeerVerification", "value": "true|false" }] }] } } }</pre>	

Possible Error Cases

- Invalid Input.
- Settings not exist
- Invalid configuration name.
- Improper configuration description.
- Unknown exception occurred.
- No operation to perform, configuration is not present.
- Local resource URL must contain protocol.
- Local resource URL details required.
- hostpeerverification for cloud server must be true.
- Invalid agentautoupgrade.
- Invalid agentinteractionmode.
- manual options for agent upgrade is not supported.
- Invalid agent auto upgrade input.

Delete Existing Configuration Settings

This method allows you to delete one or more Saner Agent configuration settings. 'accountid' should be passed as part of the following query parameter:
https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: deleteConfig

Method Type: POST

Mandatory Parameters: configname

Sample request	Sample response
<pre>{ "request": { "method": "deleteconfig", "parameters": { "parameterset": [{</pre>	In case of success: <pre>{ "response": { "method": "deleteconfig", "results": {</pre>

Sample request	Sample response
<pre>"parameter": [{ "key": "configname", "value": "myconfig" }] }] } } }</pre>	<pre>"result": [{ "key": "myconfig", "status": "SUCCESS", "reason": "" }] } } }</pre>
	In case of failure: <pre>{ "response": { "method": "deleteconfig", "results": { "result": [{ "key": "myconfig", "status": "FAIL", "reason": "Invalid configuration name" }] } } }</pre>

Possible Error Cases

- Invalid Input.
- Invalid configuration name.
- Improper configuration description.
- Unknown exception occurred.
- No operation to perform, configuration is not present.

Device Management

Using Saner, you can perform various operations on the devices in your Organization. At the same time, Saner Device Management gives you device details such as IP Address, Hostname, Mac address, and Operating System information. This section provides insights into Saner Device Management APIs and their usage.

Add Device

This method allows you to add devices. It requires IP, Hostname, MAC address, and Operating system. You are also allowed to add the corresponding device's group name. 'accountid' should be passed as part of the following query parameter: https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: addDevice

Method Type: POST

Mandatory Parameters: ip, mac, and hostname

Sample request	Sample response
<pre>{ "request": { "method": "adddevice", "parameters": { "parameterset": [{ "parameter": [{ "key": "ip", "value": "192.168.x.x" }, { "key": "hostname", "value": "new" }, { "key": "mac", "value": "99-4D-XX-XX-XX-XX" }, { "key": "os", "value": "windows 10" }, { "key": "tags", "items": [{ "key": "Location", "value": "New York" }]}]}]}}}</pre>	In case of success: <pre>{ "response": { "method":"adddevice", "results": { "result": [{ "key":"my-pc", "status":"SUCCESS", "reason":"" }] } } }</pre>
	In case of failure: <pre>{ "response": { "method":"adddevice", "results": { "result": [{ "key":"my-pc", "status":"FAIL", "reason":"Hostname exists" }] } } }</pre>

Possible Error Cases

- Failed due to invalid hostname.
- Failed due to invalid IP address.
- Failed due to invalid MAC address.
- Duplicate hostname.

Update Existing Devices

This method allows you to update device information. It requires Hostname, and you can choose to update device information such as OS or group name. You could also deactivate or reactivate agents on endpoints with agent activation (deactivate/reactivate). 'accountid' should be passed as part of the following query parameter: https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: updateDevice

Method Type: POST

Mandatory Parameters: hostname

Sample request

```
{
  "request": {
    "method": "updatedevice",
    "parameters": {
      "parameterset": [{
        "parameter": [{
          "key": "critical",
          "value": "true"
        },{
          "key": "hostname",
          "value": "my-pc"
        },{
          "key": "os",
          "value": "windows 7"
        }]
      }]
    }
  }
}
```

Request to add tag to a device:

```
{
 "request": {
  "method": "updateDevice",
  "parameters": {
   "parameterset": [
    {
     "parameter": [
      {
       "key": "hostname",
       "value": "test-laptop"
      },
      {
       "key": "tags",
       "items": [
        {
         "key": "Location",
         "value": "Bangalore"
        },
        {
         "key": "Owner",
         "value": "User_1"
        }
       ]
      }
     ]
    }
   ]
  }
 }
```

Sample response

In case of success:

```
{
  "response": {
    "method":"updatedevice",
    "results": {
      "result": [{
        "key":"my-pc",
        "status":"SUCCESS",
        "reason":""
      }]
    }
  }
}
```

In case of failure:

```
{
  "response": {
    "method":"updatedevice",
    "results": {
      "result": [{
        "key":"my-pc",
        "status":"FAIL",
        "reason":"Hostname not Found"
      }]
    }
  }
}
```

Sample request	Sample response
} Request to delete device tags { "request": { "method": "updateDevice", "parameters": { "parameterset": [{ "parameter": [{ "key": "hostname", "value": "test-laptop" }, { "key": "deletetags", "values": ["Location"] }] }] } } }	

Possible Error Cases

- No Records.
- Invalid Hostname.
- Invalid os name.
- Field Hostname cannot be empty.
- Invalid agent activation input.
- Tag value <tag_value> of key <tag_key> is invalid. Value should be either true or false.

Delete Existing Devices

This method allows you to delete one or more devices. It only requires device's hostname. If more than 15 devices are deleted, the request will be accepted, and the deletion will happen in the background. Following response would be received "<hostname> device deletion has been initiated successfully. Changes will reflect in a while." 'accountid' should be passed as part of the following query parameter: https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: deleteDevice

Method Type: POST

Mandatory Parameters: hostname

Sample request	Sample response
{ "request": { "method": "deletedevice", "parameters": {	In case of success: { "response": { "method":"deletedevice",

Sample request	Sample response
<pre>"parameterset": [{ "parameter": [{ "key": "hostname", "value": "my-pc" }] }] } } }</pre>	<pre>"results": { "result":[{ "key":"my-pc", "status":"SUCCESS", "reason":"" }] } } }</pre>
	In case of failure: <pre>{ "response": { "method":"deletedevice", "results": { "result”: [{ "key":"my-pc", "status":"FAIL", "reason":"Hostname not Found" }] } } }</pre>

Possible Error Cases

- Failed due to invalid hostname.
- Field key cannot be empty.
- Invalid host name.

Move Device

Moves one or more devices to a different group, account, or organization. This API allows administrators to move devices to a different organization, account, and group. Devices can be moved within the same organization or across organizations and accounts. Optionally, devices can be pinned to the target group upon completion of the move, ensuring that they remain associated with the selected group regardless of grouping criteria.

Method Name: movedevice

Method Type: POST

Mandatory Parameters: sourceaccount, targetaccount, targetgroup, devices, pin

Sample Request	Sample Response
<pre>{ "request": { "method": "movedevice", "parameters": { "parameterset": [{ "parameter": [{ "key": "sourceaccount", "value": "Saner-Account" }, { "key": "targetaccount", "value": " Saner-Account" }, { "key": "targetgroup", "value": "Saner-Group" }, { "key": "devices", "values": [</pre>	In case of success: <pre>{ "response": { "method": "moveDevice", "results": { "result": [{ "key": "moveDevice", "status": "SUCCESS", "reason": "Devices moved success- fully" }] } } }</pre>
	In case of failure: <pre>{ "response": {</pre>

<table>
<tr>
<td>

```
                "win-server"
                            ]},
          {
            "key": "pin",
            "value": false
                        }]
              }]
        }
    }
}
```

</td>
<td>

```
   "method": "moveDevice",
     "results": {
       "result": [{
         "key": "moveDevice",
         "status": "FAIL",
         "reason": "Group cannot be up-
dated for pinned device(s)"
                }]
        }
    }
}
```

</td>
</tr>
</table>

Pin and Unpin Devices

By default, devices are assigned to groups based on group criteria (for example: hostname, IP address, operating system, family, device type or tags). When a device is pinned to a group, it remains associated with that group regardless of group criteria. When the device is unpinned, grouping criteria are reapplied and the device is reassigned based on the configured group criteria.

Method Name: **pinandupindevice**

Method Type: POST

Mandatory Parameters: action, devices, accountname

To pin device:

<table>
<tr><th>Sample Request</th><th>Sample Response</th></tr>
<tr>
<td rowspan="2">

```
{
  "request": {
    "method": "pinandupindevice",
      "parameters": {
        "parameterset": [{
           "parameter": [{
               "key": "action",
               "value": "pin"
                        },
          {
            "key": "devices",
            "values": [
               "windows-machine "
                        ]},
          {
             "key": "accountname",
              "value": "Saner-Account "
                        }
                    ]}
             ]}
    }
}
```

</td>
<td>

In case of success:

```
{
  "response": {
    "method": "pinandupindevice",
    "results": {
      "result": [{
           "key": "pinUnpinDevices",
           "status": "SUCCESS",
           "reason": "Devices pinned suc-
cessfully"
                }]
        }
    }
}
```

</td>
</tr>
<tr>
<td>

In case of failure:

```
{
  "response": {
    "method": "pinandupindevice",
    "results": {
      "result": [{
         "key": "devices",
          "status": "FAIL",
          "reason": "No valid devices
found to pin"
                }]
        }
    }
}
```

</td>
</tr>
</table>

To unpin a device:

Sample Request	Sample Response
<pre>{ "request": { "method": "pinandupindevice", "parameters": { "parameterset": [{ "parameter": [{ "key": "action", "value": "unpin" }, { "key": "devices", "values": ["win10", "winserver"]}, { "key": "accountname", "value": "Saner-Account" }] }] } } }</pre>	In case of success: <pre>{ "response": { "method": "pinandupindevice", "results": { "result": [{ "key": "pinUnpinDevices", "status": "SUCCESS", "reason": "Devices unpinned successfully" }] } } }</pre> In case of failure: <pre>{ "response": { "method": "pinandupindevice", "results": { "result": [{ "key": "action", "status": "FAIL", "reason": "Invalid action. Allowed actions are pin and unpin" }] } } }</pre>

Get Devices Information

This method allows you to get device information. It requires one of the following parameters to fetch results – Hostname, IP, MAC Address, Serial Number or tagname and tagvalue". 'accountid' should be passed as part of the following query parameter:
https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: **getDevice**

Method Type: POST

Mandatory Parameters: hostname

Sample request	Sample response

To get device information using a hostname:

```
{
  "request": {
    "method": "getdevice",
    "parameters": {
      "parameterset": [{
        "parameter": [{
          "key": "hostname",
          "value": "my-pc"
        }]
      }]
    }
  }
}
```

To get device information using an IP or MAC address:

```
{
  "request": {
    "method": "getdevice",
    "parameters": {
      "parameterset": [{
        "parameter": [{
          "key": "ip",
          "value": "192.168.XX.XX"
        }]
      },{
        "parameter": [{
          "key": "mac",
          "value": "88-77-99-XX-XX"
        }]
      }]
    }
  }
}
```

To get information based on tags:

```
{
 "request": {
  "method": "getdevice",
  "parameters": {
   "parameterset": [
    {
     "parameter": [
      {
       "key": "apifilters",
       "filters": [
        {
         "tags": {
          "in": {
           "ip_address": [
            "172.16.XX.XXX"
           ]}}}]}]}}}}
```

In case of success(For Saner Agent installed device):


```
{
 "devices": {
  "device": [
   {
    "majorVersion": "6.3",
    "minorVersion": "0.0-noui",
    "arch": "x86",
    "type": "exe",
    "release": "1",
    "lastseen": "Active",
    "family": "windows",
    "deviceType": "end-point",
    "scannedBy": "Scanner",
    "networkScanConfig": "",
    "lastScan": "2024-12-22 22:33:00 (UTC-08:00)",
    "manufacturer": "",
    "serialNumber": "Not Specified",
    "sysinfo_osname": "Microsoft Windows 10",
    "sticky": true,
    "remoteAccessToolDownload": false,
    "lastRemediation": "2024-12-22 22:27:00 (UTC-08:00)",
    "agentStatus": "ScanDone",
    "critical-resource": "false",
    "group-id": "windows 10",
    "profile-id": "Test",
    "host-name": "my-pc",
    "ip-address": "172.16.XX.XX",
    "mac-address": "2E-A4-XX-XX-31-2D",
    "operating-system": "Microsoft Windows 10 v22H2 architecture 64-bit",
    "host-reachable": "true",
    "agent-version": "6.3;0.0-noui;1;exe;x86",
    "agent-enabled": "76a105de-XXXX-4aXX-ba0a-fc680XXXX818",
    "customId": "",
    "time-stamp": "1734935706519",
    "creation": "1733118539594",
    "tags": [
     {
      "key": "ip_address",
      "value": "172.16.XX.XX"
     },
     {
      "key": "mac_address",
      "value": "2E-XX-XX-XX-XX-2D"
     },
     {
      "key": "name",
      "value": "my-pc"
     },
     {
      "key": "os",
      "value": "Microsoft Windows 10 v22H2 architecture 64-bit"
     },
     {
      "key": "subcategory",
      "value": "end-point"
     },
     {
      "key": "group",
      "value": "windows 10"
     }
    ],
    "agentConfig": "debug"
   }
  ]
 }
}
```

In case of success(For Network device):


```
{
 "devices": {
```

```
  "device": [
   {
    "majorVersion": "",
    "minorVersion": "",
    "arch": "",
    "type": "",
    "release": "",
    "lastseen": "active",
    "family": "others",
    "deviceType": "general purpose",
    "scannedBy": "Scanner",
    "networkScanConfig": "test",
    "lastScan": "2024-12-03 04:33:00 (UTC-
08:00)",
    "manufacturer": "",
    "serialNumber": "Not Specified",
    "sysinfo_osname": "Linux 5.10 - 5.18",
    "sticky": false,
    "remoteAccessToolDownload": false,
    "lastRemediation": "",
    "agentStatus": "",
    "group-id": "general purpose",
    "host-name": "172.16.XX.XX",
    "ip-address": "172.16.XX.XX",
    "mac-address": "BC-24-XX-XX-XX-A9",
    "operating-system": "Linux 5.10 - 5.18",
    "host-reachable": "true",
    "agent-version": "",
    "agent-enabled": "f3b196b6-XXXX-41d1-XXXX-
f4480c4fcf24",
    "time-stamp": "1733229207190",
    "creation": "1733122441525",
    "tags": [
     {
      "key": "ip_address",
      "value": "172.16.XX.XX"
     },
     {
      "key": "mac_address",
      "value": "BC-24-XX-XX-03-A9"
     },
     {
      "key": "name",
      "value": "172.16.XX.XX"
     },
     {
      "key": "os",
      "value": "Linux 5.10 - 5.18"
     },
     {
      "key": "subcategory",
      "value": "general purpose"
     },
     {
      "key": "group",
      "value": "general purpose"
     }
    ]
   }
  ]
 }
}
```


Possible Error Cases

- No Records
- Invalid Serial Number.
- Invalid Mac address.

Get Device Details

This method is used to get a given device's vulnerability and compliance details.

Method Name: getDeviceDetails

Method Type: POST

Mandatory Parameters: accountid and hostname

Sample request	Sample response
<pre>{ "request": { "method": "getDeviceDetails", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "test_account" }, { "key": "hostname", "value": "test_machine" }, { "key": "apifilters", "filters": [{ "tags": {</pre>	<pre>API response will be in .zip format</pre>

Sample request	Sample response
<pre>"in": { "owner": ["User_1"] }, "not in": { "owner": ["User_1"] } } }] }] }] } } }	

Possible Error Cases

- Invalid Input
- Invalid Account Name
- Invalid Hostname
- Invalid IP Address
- Invalid MAC address
- Invalid Tag name
- Invalid Tag value

Get Device Vulnerabilities

This method returns all the vulnerabilities that exist on a device. You can get the vulnerabilities for a single device or all devices. If you don't specify the hostname, vulnerabilities present on all the devices in the Account gets listed. To fetch vulnerabilities for all network devices, set the deviceType key to "network" and omit the hostnames key. Similarly, to fetch vulnerabilities for all endpoint devices, set the deviceType key to "end-point" and omit the hostnames key." Refer sample request for more details.

Method Name: getDeviceVulnerabilities

Method Type: POST

Mandatory Parameters: accountname

Sample request	Sample response
<pre>{ "request": { "method": "getDeviceVulnerabilities", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountname", "value": "Test_Account"</pre>	<pre>In case of success: [{ "hostname": "test-machine", "deviceType": "end-point", "result": [{ "assetname": "OpenSSH", "platform": "cpe:/o:remote_detect:remote_detect",</pre>

Sample request	Sample response
<pre>}, { "key": "hostnames", "values": ["test-machine"] }] }] } }</pre> To get vulnerabilities based on devicetype: <pre>{ "request": { "method": "getDeviceVulnerabilities", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountname", "value": "_Default" }, { "key": "hostnames", "values": [] }, { "key": "deviceType", "value": "end-point" }] }] } }</pre>	<pre>"product": "cpe:/a:openbsd:openssh", "concerns": [{ "id": "CVE-2007-2768", "type": "VULNERABILITY", "title": "Authentication Bypass Vulnerability in OpenSSH with OPIE for PAM", "score": 4.3, "severity": "Medium", "portservice": "22/ssh", "evidence": "The vulnerability exists because the device is running OpenSSH 7.3 which is vulnerable", "detectiondate": "2024-07-26", "releasedate": "2007-05-21", "fixinfo": "Please access the vendor's website by clicking on the following link: \nhttps://www.openbsd.org/ \n \nOnce on the website, download the most up-to-date version of OpenSSH. \nYou can also download it from the following link: \nhttps://www.openssh.com/ \n \nInstall OpenSSH following the steps from the link: \nhttps://www.tecmint.com/install-openssh-server-in-linux/ \n \nPrerequisites: After successfully installing the latest version of OpenSSH, it is essential to note that the previous version may remain on your system. It is recommended to uninstall the earlier version of OpenSSH manually." }, { "id": "CVE-2008-3844", "type": "VULNERABILITY", "title": "Trojan Horse Vulnerability in Certain Red Hat Enterprise Linux packages", "score": 9.3, "severity": "Critical", "portservice": "22/ssh", "evidence": "The vulnerability exists because the device is running OpenSSH 7.3 which is vulnerable", "detectiondate": "2024-07-26", "releasedate": "2008-08-27", "fixinfo": "Please access the vendor's website by clicking on the following link: \nhttps://www.openbsd.org/ \n \nOnce on the website, download the most up-to-date version of OpenSSH. \nYou can also download it from the following link: \nhttps://www.openssh.com/ \n \nInstall OpenSSH following the steps from the link: \nhttps://www.tecmint.com/install-openssh-server-in-linux/ \n \nPrerequisites: After successfully installing the latest version of OpenSSH, it is essential to note that the previous version may remain on your system. It is</pre>

Sample request	Sample response
	recommended to uninstall the earlier version of OpenSSH manually." }], "riskscount": 2 }, { "assetname": "Microsoft IIS", "platform": "cpe:/o:remote_detect:remote_detect", "product": "cpe:/a:microsoft:internet_information_services", "riskscount": 0 }] }]

Possible Error Cases

- Missing mandatory fields. Account name must be provided
- Field key cannot be empty
- Invalid host name {host name}
- Failed due to invalid account name or Id
- Field key is invalid
- Account name must be provided
- {host names} are not {device type} devices
- Invalid device type. It can be either end-point or network
- Failed to get vulnerabilities of the given devices
- {host names} do not exist

Get Account Information

This method allows you to get an account name for a given subscription id. To get the subscription id, run the getHostResults. The response of the getHostResults API contains the subscription id. 'accountid' should be passed as part of the following query parameter:
https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: **getDeviceAccountInfo**

Method Type: POST

Mandatory Parameters: subscriptionid

Sample request	Sample response
{ "request": { "method": "getdeviceaccountinfo", "parameters": { "parameterset": [{ "parameter": [{ "key": "subscriptionid",	In case of success: { "response": { "method": "getdeviceaccountinfo", "results": { "result": [{ "accountname": "Test Account" }]

Sample request	Sample response
<pre>"value": "2edbea6f-c022-4302- 972d-a11e80e7284d" }] }] } } }</pre>	<pre>} } } In case of failure: { "response": { "method": "getdeviceaccountinfo", "results": { "result": [{ "key": "subscriptionid", "status": "FAIL", "reason": "Subscription id not found" }] } } }</pre>

Possible Error Cases

- No devices present for given Subscription ID
- Account has expired.
- Invalid Input

Scan Devices

This method is used to start a scan on endpoints. 'accountid' should be passed as part of the following query parameter: https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: scanDevice

Method Type: POST

Mandatory Parameters: hostname

Sample request	Sample response
<pre>{ "request": { "method": "scandevice", "parameters": { "parameterset": [{ "parameter": [{ "key": "hostname", "value": "desktop-ujk5rbn" },{ "key": "ip", "value": "192.168.XX.XX" },{ "key": "mac", "value": "00-0C-29-XX-XX-XX"</pre>	<pre>In case of success: { "response": { "method": "scandevice", "results": { "result": [{ "key": "null", "status": "SUCCESS", "reason": "" }] } } }</pre>

Sample request	Sample response
<pre>}] }] } } }</pre>	

Possible Error Cases

- Invalid Host Name.
- Invalid IP Address.
- Invalid MAC Address.

Get Device Reports

Saner Agents upload endpoints' reports onto the Saner Server after each scan. This method allows you to fetch reports of the endpoints. Support added to fetch results with tagname and tagvalue, and includeDSI. To retrieve a device report with Detailed System Information, you need to pass the key 'includeDSI' and specify its value as true. By default, the value for key'includeDSI' is set to *false*. 'accountid' should be passed as part of the following query parameter:
https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method name: getDeviceReport

Method Type: POST

Mandatory Parameters: hostname

Sample request	Sample response
To get a device report using its hostname: <pre>{ "request": { "method": "getdevicereport", "parameters": { "parameterset": [{ "parameter": [{ "key": "hostname", "value": "my-pc" },{ "key": "tagname", "value": "tag01" },{ "key": "includeDSI", "value": "false" }, { "key": "tagvalue", "value": "111-222-333-444" }] }] } } }</pre>	The API response will be in .zip file format. The zip file contains two JSON files – one for all the vulnerabilities on the device and one for all the misconfigurations on the device.
To get a device report using an IP or MAC address:	

Sample request	Sample response
<pre>{ "request": { "method": "getdevicereport", "parameters": { "parameterset": [{ "parameter": [{ "key": "ip", "value": "192.168.XX.XX" }] },{ "key": "includeDSI", "value": "false" },{ "parameter": [{ "key": "mac", "value": "88-77-XX-XX-XX" }] }] } } }</pre>	
To get reports of all devices: <pre>{ "request": { "method": "getdevicereport”, "parameters": {} } }</pre>	
To get device report with tagname and tagvalue: <pre>{ "request": { "method": "getdevicereport", "parameters": { "parameterset": [{ "parameter": [{ "key": "tagname", "value": "tag01" },{ "key": "tagvalue", "value": "111-222-333-444" },{ "key": "includeDSI", "value": "false" },{ "key": "hostname", "value": "my-pc" }] }] } } }</pre>	
To get device report with Details System Information <pre>{ "request": { "method": "getdevicereport", "parameters": { "parameterset": [{ "parameter":{ "key": "includeDSI", "value": "true"</pre>	

Sample request	Sample response
<pre>},{ "key": "hostname", "value": "my-pc" }] }] } } }</pre>	

Possible Error Cases

- Invalid 'includeDSI' value.
- No Devices Found
- No Records Found

Get Basic System Details

This method lets you get the basic system details of the devices in an Account. Details such as the IP address, MAC address, memory, disk size, serial number, systemUUID, CPU information, and operating system details are provided in the response when the 'getBasicSystemDetails' API is executed.

The 'getBasicSystemDetails' API will fetch details of all the devices in the Account if the key 'hostnames' and its corresponding value are not mentioned in the request body.

Method Name: getBasicSystemDetails

Method Type: POST

Mandatory Parameters: accountname

Sample request	Sample response
<pre>{ "request": { "method": "getBasicSystemDetails", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountname", "value": "Test_Account1" }] }] } } }</pre>	<pre>{ "hostDetails": [{ "name": "515-sys-win2019", "systemName": "515-SYS-WIN2019", "ipAddress": "172.16.X.X", "macAddress": "BC-XX-XX-XX-XX-XX", "sanerEnabled": "true", "deviceType": "end-point", "lastLogonUsername": "515-SYS-WIN2019\\Administrator", "ram": "8.0 GiB", "diskSize": "100.0 GiB", "cpuSpeed": "", "cpuCores": "4", "serialNumber": "", "systemUUID": "30F9XXXX-XXXX-XXXX-XXXX-1A65220EXXXX", "cpuArchitecture": "x64", "systemProductName": "Standard PC (Q35 + ICH9, 2009)", "systemDisplay": "true", "osName": "Microsoft Windows Server 2019 Standard", "osVersion": "10.0.17763.678", "osServicePack": "", "defaultGateway": "172.16.X.X",</pre>

```
   "tags": [
    {
     "key": "cpu_name",
     "value": "Common KVM processor"
    },
    {
     "key": "disk_space",
     "value": "100.0 GiB"
    },
    {
     "key": "manufacturer",
     "value": "QEMU"
    },
    {
     "key": "group",
     "value": "windows server 2019"
    },
    {
     "key": "name",
     "value": "515-sys-win2019"
    },
    {
     "key": "mac_address",
     "value": "BC-XX-XX-XX-XX-XX"
    },
    {
     "key": "subcategory",
     "value": "end-point"
    },
    {
     "key": "cpu_core_count",
     "value": "4"
    },
    {
     "key": "ip_address",
     "value": "172.16.X.X"
    },
    {
     "key": "os",
     "value": "Microsoft Windows Server 2019
v1809 architecture 64-bit"
    },
    {
     "key": "ram",
     "value": "8.0 GiB"
    }
   ]
  }
 ]
}
```


Possible Error Cases

- Access Denied
- Missing required parameters. Account must be provided.
- Field <key> cannot be empty.
- Failed due to invalid account name or ID
- Field accountname cannot be empty.
- No hosts found.
- Hostname cannot be empty.
- Failed due to invalid hostname: <hostname>
- Host <hostname> does not exist.
- No Records.

Get Device Tag Keys

This method retrieves all the tags assigned to devices. Using the 'getDeviceTags' API, you can fetch tags assigned to individual devices and for all the devices in an Account.

Method Name: getDeviceTagKeys

Method Type: POST

Mandatory Parameters: accountname

Sample Request

```
{
 "request": {
  "method": "getDeviceTagKeys",
  "parameters": {
   "parameterset": [
    {
     "parameter": [
      {
       "key": "accountnames",
       "values": [
        "Test_Account_1",
        "Test_Account_2"
       ]
      }
     ]
    }
   ]
  }
 }
}
```

Sample Response

```
{
    "tagDetails": [
        {
            "tags": [
                "new",
                "cpu_name",
                "color",
                "os",
                "os_version",
                "internet_facing",
                "team",
                "ip_address",
                "disk_space",
                "devtype",
                "mac_address",
                "data_centric",
                "name",
                "business_centric",
                "location",
                "hello",
                "subcategory",
                "cpu_core_count",
                "group",
                "ram"
            ],
            "accountName": "Test_Ac-
count_1"
        },
        {
            "tags": [],
            "accountName": "Test_Ac-
count_2"
        }
    ]
}
```

Possible Error Cases

- Access Denied
- Missing required parameters. Account must be provided.
- Field <key> cannot be empty.
- Failed due to invalid account name or ID
- Field accountname cannot be empty.
- No hosts found.
- Hostname cannot be empty.
- Failed due to invalid hostname: <hostname>
- Host <hostname> does not exist.
- No Records.

Get Installed Applications

This method gets all the installed applications on the devices that exist in an Account. Details such as the application name, version, publisher, and host details are returned as a response when the 'getInstalledApplications' API is executed.

The 'getInstalledApplications' API will fetch applications installed on all the devices in the Account if the key 'hostnames' and its corresponding value are not mentioned in the request body.

Method Name: getInstalledApplications

Method Type: POST

Mandatory Parameters: accountname

Sample request	Sample response
<pre>{ "request": { "method": "getInstalledApplications", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountname", "value": "Test_Account" }, { "key": "hostnames", "values": ["testmachine-1"] }] }] } } }</pre>	<pre>{ "applicationDetails": [{ "hosts": ["testmachine-1"], "name": "Microsoft.Xbox.TCUI", "version": "1.24.10001.0", "publisher": "Microsoft Corporation" }, { "hosts": ["testmachine-1"], "name": "Microsoft.WindowsCamera", "version": "2021.105.10.0", "publisher": "Microsoft Corporation" }, { "hosts": ["testmachine-1"], "name": "Microsoft.ScreenSketch", "version": "10.2008.3001.0", "publisher": "Microsoft Corporation" }] }</pre>

Possible Error Cases

- Access Denied
- Missing required parameters. Account name must be provided.
- Field <key> cannot be empty.
- Failed due to invalid account name or Id.
- Field accountname cannot be empty.
- No hosts found.
- Hostname cannot be empty.
- Failed due to invalid hostname : <hostname>
- Host <hostname> does not exist.

- No Records.

Get Device Job Information

This method will get all the jobs created for a host. This includes Remediation (Rem Job), IR (Non-Security, Reboot, Rollback, Firmware) and Automation rule Jobs. Refer to Job type and Creation date. Required field is Hostname and Accountname.

Method Name: **getDeviceJobInfo**

Method Type: POST

Mandatory Parameters: hostname and accountname

Sample request	Sample response
<pre>{ "request": { "method": "getDeviceJobInfo", "parameters": { "parameterset": [{ "parameter": [{ "key": "hostname", "value": "Test-Desktop" }, { "key": "accountname", "value": "Demo Account" }, { "key": "tool", "values": ["PM"] }] }] } } }</pre>	<pre>In case of success: { "response": { "method": "getDeviceJobInfo", "results": { "result": [{ "hostalljobsdetails": [{ "RuleName": "changedName", "CreatedBy": "cvemtest", "CreationTime": "2025-07-04 07:50:44 AM UTC", "Groupname": "Prod_Win_Devices", "Hostname": "Test-Desktop", "Asset": "", "PatchName": "", "patchSize": "", "InstalledVersion": "", "Oldversion": "", "risksmitigated": "", "risksmitigatedcount": "", "Status": "up-to-date", "Reason": "", "LastUpdate": "2025-07-04 07:50:44 AM UTC", "jobType": "Automation Rule" }, { "JobName": "Job_1750247389199", "CreationTime": "2025-06-18 11:49:48 AM UTC", "Groupname": "Prod_Win_Devices", "Hostname": "Test-Desktop", "Asset": "WinSCP x86", "PatchName": "WinSCP-6.5.1-Setup-x86.exe", "patchSize": "11730302", "InstalledVersion": "6.5.1", "Oldversion": "Unknown", "risksmitigated": ["CVE-2021-3331", "CVE-2024-31497", "CVE-2023-48795"], "risksmitigatedcount": 3, "Status": "success", "Reason": "The remediation task installed successfully.", "LastUpdate": "2025-06-18 11:59:13 AM UTC", "CreatedBy": "cvemtest", "jobType": "Rem Job" }</pre>

Sample request	Sample response
	] }] } } } In case of failure such as Invalid Hostname { "response": { "method": "getDeviceJobInfo", "results": { "result": [{ "key": "hostname", "status": "FAIL", "reason": "Host not found" }] } } }

Possible Error Cases

- Hostname does not exist.
- Accountname does not exist.
- Account has expired.
- Invalid Input.

Get Host Results

This method provides detailed information of a particular device including network interface details. 'accountid' should be passed as part of the following query parameter:
https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: getHostResults

Method Type: POST

Mandatory Parameters: SEARCH field now accepts the following fields:

IP, SUBID, MACADDR, HOSTNAME, SEARCH, and TAGS

Sample request	Sample response
{ "request": { "method": "gethostresults", "parameters": { "parameterset": [{ "parameter": [{ "key": "SEARCH", "value": "192.198.XX.XX" }] }] } }	{ "agentResults": [{ "results": { "metadata": { "last-remediation": "2025-07-10 12:12:00 (UTC- 07:00)", "last-scan": "2025-07-11 12:05:00 (UTC-07:00)", "last-update": "2025-07-11 11:00:00 (UTC-07:00)", "next-scan": "2025-07-11 12:00:00 (UTC-07:00)", "next-update": "2025-07-12 11:00:00 (UTC-07:00)", "vulnerabilities-total-count": "2579", "vulnerabilities-count": "2579", "severity-status": "low-165 medium-1462 high-719

Sample request	Sample response
}	critical-233 total-17033.4", "severity-total": 17033.4, "severity-critical": 233, "severity-high": 719, "severity-medium": 1462, "severity-low": 165 }, "sysinfo": { "os_release_id": "21H2", "agent_profile_id": "ad3babc5-3c22-4dc7-9cdf-255424aac328", "architecture": "64-bit", "interfaces": { "interface": [{ "interface_name": "Intel(R) PRO/1000 MT Network Connection", "ip_address": "172.16.X.X", "mac_address": "BC-24-XX-XX-XX-E4", "gateway": "", "type": "", "broadcast_address": "", "netmask": "" }] }, "family": "windows", "os_name": "Microsoft Windows 11", "os_version": "6.3.22000", "primary_host_name": "Test-Desktop", "agent_version": "6.5;0.0-noui;1;exe;x86", "subs_mac_addr": "BC-24-XX-XX-XX-E4", "primary_host_ip": "172.16.X.X", "primary_host_mac": "BC-24-XX-XX-XX-E4", "subs_id": "1f1XXXX-XXXX-XXXX-XXXX-cXXXXX" } } }] }

Possible Error Cases

- No data found
- Account not found
- Account cannot be empty

Get Device Job Summary

This method feature is used to retrieve job summary details for a given host.

Method Name: getDeviceJobsummary

Method Type: POST

Mandatory Parameters: hostname and accountname

Sample request	Sample response
{	{

Sample request	Sample response
<pre>"request": { "method": "getDeviceJobSummary", "parameters": { "parameterset": [{ "parameter": [{ "key": "hostname", "value": "desktop1" }, { "key": "accountname", "value": "_Default.TestOrg" }, { "key": "tool", "values": ["PM", "CM"] }] }] } }}</pre>	<pre>"response": { "method": "getdevicejobsummary", "results": { "result": [{ "devicejobsummary": [{ "JobName": "sec_patch_01", "OverallStatus": "not applicable", "JobType": "Rem Job", "Tool": "PM" }, { "JobName": "PM_rule_01", "OverallStatus": "fail", "JobType": "Automation Rule", "Tool": "PM" }, { "JobName": "test_roll_02", "OverallStatus": "Received", "JobType": "IR", "Tool": "PM" }] }]} }}</pre>

Possible Error Cases

- Invalid Input.
- Invalid Hostname.
- Invalid Accountname.
- Invalid Tool.
- Not enough inputs found.
- Account not found.
- No service provisioned tools found for entered account.
- PM tool is not enabled for entered account.
- CM tool is not enabled for entered account.
- No supported tools found in the API input.
- Host not found.
- No jobs found for host.

Get Device Job Details

This method is used to retrieve all the job details created for a host in a nested model.

Method Name: getDeviceJobDetails

Method Type: POST

Mandatory Parameters: hostname and accountname

Sample request	Sample response
<pre>{ "request": { "method": "getdevicejobdetails", "parameters": { "parameterset": [{ "parameter": [{ "key": "hostname", "value": "ubuntu1804" },</pre>	<pre>{ "response": { "method": "getdevicejobdetails", "results": { "result": [{ "devicejobdetails": [{ "JobName": "sec_job01", "CreationDate": "2023-07-04 16:06:56",</pre>

Sample request	Sample response
<pre>{ "key": "accountname", "value": "_Default.TestOrg" }, { "key": "tool", "value": "PM" }, { "key": "jobname", "value": "sec_job01" }, { "key": "jobtype", "value": "Rem Job" }] }] } } }	<pre>"HostName": "ubuntu1804", "Group": "ubuntu", "OverallStatus": "success", "LastUpdate": "2023-07-04 16:09:39", "Patches": [{ "Asset": "libxml2", "PatchName": "libxml2:amd64", "PatchSize": "", "patchSize": "648.2 KiB", "InstalledVersion": "2.9.4+dfsg1-6.1ubuntu1.6", "OldVersion": "2.9.4+dfsg1-6.1ubuntu1.2", "Status": "success", "RisksMitigated": ["CVE-2019-19956", "CVE-2020-7595", "CVE-2019-20388", "CVE-2020-24977", "CVE-2021-3516", "CVE-2021-3517", "CVE-2021-3518", "CVE-2021-3537"], "RisksMitigatedCount": 8, "Reason": "" }] }] }] } }</pre>

Possible Error Cases

- Invalid Input.
- Invalid Hostname.
- Invalid Accountname.
- Invalid Tool.
- Not enough inputs found.
- Account not found.
- No service provisioned tools found for entered account.
- PM tool is not enabled for entered account.
- CM tool is not enabled for entered account.
- Entered tool is not supported for this API.
- Host not found.
- No jobs found for host.

Update Device Alias

This method updates the device name to a given custom name that will be displayed in Saner web console. 'accountid' should be passed as part of the following query parameter:
https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: updateDeviceAlias

Method Type: POST

Mandatory Parameters: hostname

Sample request	Sample response
<pre>{ "request": { "method": "updateDeviceAlias", "parameters": { "parameterset": [{ "parameter": [{ "key": "hostname", "value": "my-pc" },{ "key": "alias", "value": "my-pc-aliased" }] }] } } }</pre>	In case of success: <pre>{ "response":{ "method":"updateDeviceAlias", "results":{ "result":[{ "key":"my-pc", "status":"SUCCESS", "reason":"" }] } } }</pre>
	In case of failure: <pre>{ "response":{ "method":"updateDeviceAlias", "results":{ "result":[{ "key":"my-pc", "status":"FAIL", "reason":"This is a duplicate entry." }] } } }</pre>

Possible Error Cases

- Duplicate entry.
- Invalid host name.

Remove Device Alias

This method deletes the custom name provided to a device. ‘accountid’ should be passed as part of the following query parameter: https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: removeDeviceAlias

Method Type: POST

Mandatory Parameters: alias

Sample request	Sample response
<pre>{ "request": { "method": "removeDeviceAlias", "parameters": { "parameterset": [{ "parameter": [{ "key": "alias", "value": "my-pc-aliased" }] }] } }</pre>	In case of success: <pre>{ "response":{ "method": "removeDeviceAlias", "results":{ "result":[{ "key":"my-pc-aliased", "status":"SUCCESS", "reason":"" }] } }</pre>

Sample request	Sample response
}	}
	{ "response":{ "method":"removeDeviceAlias", "results":{ "result":[{ "key":"my-pc-aliased", "status":"FAIL", "reason":"No device found in the given account with given details." }] } } }

Possible Error Cases

- No device found.
- Field key cannot be empty
- Invalid host name

Uninstall Saner Agent from a System

This method allows you to uninstall the agent from your device. 'accountid' should be passed as part of the following query parameter: https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: uninstallAgent

Method Type: POST

Mandatory Parameters: hostname

Sample Request	Sample Response
{ "request": { "method": "uninstallagent", "parameters": { "parameterset": [{ "parameter": [{ "key": "hostname", "value": "windows-laptop" }] }] } }	{ "response": { "method": "uninstallagent", "results": { "result": [{ "key": "null", "status": "success", "reason": "" }] } } }

<pre>} } }</pre>	

Possible Error Cases:

- Invalid Input.
- No records.
- Account not found.
- Invalid Hostname.
- Invalid IP Address.
- Invalid MAC address.
- Invalid tag name.
- Invalid tag value.
- No records.

Cyber Hygiene Score

Saner Cyber Hygiene Score assigns a score to Organizations, Accounts, and Devices within the Saner ecosystem. A Cyber Hygiene Score is the quantification of the total attack surface of a device that includes Common Vulnerabilities and Exposures (CVEs), Common Configuration Enumeration(CCEs), Missing Patches, and Posture Anomalies.This section provides insights into the Saner Cyber Hygiene Score APIs and their usage.

Get Account Hygiene Score

This method fetches the cyber hygiene score of the given account.

Method Name: **getAccountHygieneScore**

Method Type: GET

Mandatory Parameters: account_name

URL: https://saner.secpod.com/CHScanner/getAccountHygieneScore

Sample request	Sample response
<pre>{saner.secpod.com}/CHScanner/getAccountHygieneScore?account_name=accountname</pre>	<pre>{ "chsresponse": [{ "account_id": "sp17f77fwXXXXXX", "account_name": "Test Account", "account_score": 18.15, "anomaly_score": 79.2,</pre>

Sample request	Sample response
	<pre>"ccss_score": 1.5, "cvss_score": 0.43, "missing_patch_score": 0.72, "no_of_devices": 83, "timestamp": "2023-05-24 05:27:40.450122" }] }</pre>

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.

Calculate Hygiene Score

This method allows you to calculate the Hygiene Score for the given account.

Method Name: calculateHygieneScore

Method Type: POST

Mandatory Parameters: account_name

URL: https://saner.secpod.com/CHScanner/calculateHygieneScore

Sample request	Sample response
<pre>{ "account_name": "account name" }</pre>	<pre>{ "message": "Successful" } If the CHS Scan is ongoing, one of the below responses will be displayed depending on the queue. {"message": "Scan has been queued"} OR {"message":"Scan Already Queued"}</pre>

Possible Error Cases

- Bad Request.
- Account not found or expired.

Get CHScoreSummary For Device With Status

This method fetches the cyber hygiene score, Saner Agent status, and hostnames for the given devices found in the Account.

Method Name: getCHScoreSummaryForDeviceWithStatus

Method Type: POST

Mandatory Parameters: account_name

URL: https://saner.secpod.com/CHScanner/getCHScoreSummaryForDeviceWithStatus

Sample request


```
{
 "account_name": "Test_Account",
 "devices": [],
 "fields": [
  "family",
  "tags"
 ],
 "search": "test-machine",
 "status": "active",
 "family": [
  "windows"
 ],
 "os": [],
 "score-category": [
  "low"
 ],
 "tags": [
  {
   "condition": "[=] is one of",
   "key": "location",
   "value": [
    "New York"
   ]
  }
 ]
}
```


Sample response


```
{
 "chsresponse": [
  {
   "agentStatus": {
    "ancorinteraction": "live",
    "devicestatus": "ScanDone",
    "displaytext": "Active",
    "lastseen": "Active",
    "rebootstatus": "false"
   },
   "family": "windows",
   "hostname": "test-machine",
   "score": 24.01,
   "tags": [
    {
     "key": "location",
     "value": "New York"
    },
    {
     "key": "ip_address",
     "value": "192.168.X.X"
    },
    {
     "key": "mac_address",
     "value": "C6-XX-XX-XX-XX-02"
    },
    {
     "key": "name",
     "value": "test-machine"
    },
    {
     "key": "os",
     "value": "Microsoft Windows 10 v22H2
architecture 32-bit"
    },
    {
     "key": "serial_number",
     "value": "Not Specified"
    },
    {
     "key": "subcategory",
     "value": "end-point"
    },
    {
     "key": "group",
     "value": "windows 10"
    },
    {
     "key": "disk_space",
     "value": "100.0 GiB"
    },
    {
     "key": "cpu_name",
     "value": "Intel(R) Xeon(R) CPU
X5650  @ 2.67GHz"
    },
    {
     "key": "os_version",
     "value": "10.0.19045.3930"
    },
    {
     "key": "ram",
     "value": "3.0 GiB"
```

Sample request	Sample response
	<pre>{ }, { "key": "cpu_core_count", "value": "2" }] }] }</pre>

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.
- No Provision.

Get CHScore Summary For Family

This method returns the cyber hygiene score, hostnames, IP addresses, local score, global score, and OS Group per family.

Method Name: getCHScoreSummaryForFamily

Method Type: POST

Mandatory Parameters: account_name

URL: https://saner.secpod.com/CHScanner/getCHScoreSummaryForFamily

Sample request	Sample response
<pre>{ "account_name": "{{account_name}}", "family": ["windows", "unix", "macos"], "devices": ["sp-test-laptop", "sp- test2-desktop"], "group": ["windows 10", "ubuntu"], "os": ["Microsoft Windows 10", "Ubuntu 20.04"] }</pre>	<pre>{ "chsresponse": [{ "family": "windows", "global_score": 1.2, "group": ["windows 10"], "hostname": ["sp-test-laptop"], "ip_address": ["192.168.x.x"], "local_score": 92.21, "os": ["Microsoft Windows 10"], "score": 19.4 }, { "family": "unix", "global_score": 1.2, "group": ["ubuntu"], "hostname": [</pre>

Sample request	Sample response
	"sp-test2-desktop"], "ip_address": ["192.168.x.x"], "local_score": 46.17, "os": ["Ubuntu 20.04"], "score": 10.19 }] }

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.
- No Provision.

Get CHScore Summary For OS

This method returns the cyber hygiene score, hostnames, IP addresses, local score, global score, family, and group per OS.

Method Name: getCHScoreSummaryForOS

Method Type: POST

Mandatory Parameters: account_name

URL: https://saner.secpod.com/CHScanner/getCHScoreSummaryForOs

Sample request	Sample response
{ "account_name": "{{account_name}}", "family": ["windows", "unix", "macos"], "devices": ["sp-test-laptop", "sp- test1-desktop"], "group": ["windows 10", "ubuntu"], "os": ["Microsoft Windows 10", "Ubuntu 20.04"] }	{ "chsresponse": [{ "family": "unix", "global_score": 1.2, "group": ["ubuntu"], "hostname": ["sp-test2-desktop"], "ip_address": ["192.168.x.x"], "local_score": 46.17, "os": "Ubuntu 20.04", "score": 10.19 }, { "family": "windows",

Sample request	Sample response
	"global_score": 1.2, "group": ["windows 10"], "hostname": ["sp-test-laptop"], "ip_address": ["192.168.x.x"], "local_score": 92.21, "os": "Microsoft Windows 10", "score": 19.4 }] }

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.
- No Provision.

Get CHScore Summary For Group

This method returns the hygiene score, hostnames, IP addresses, local score, global score, family, and OS per group.

Method Name: **getCHScoreSummaryForGroup**

Method Type: POST

Mandatory Parameters: account_name

URL: https://saner.secpod.com/CHScanner/getCHScoreSummaryForGroup

Sample request	Sample response
{ "account_name": "{{account_name}}", "family": ["windows", "unix", "macos"], "devices": ["sp-test-laptop", "sp- test1-desktop"], "group": ["windows 10", "ubuntu"], "os": ["Microsoft Windows 10", "Ubuntu 20.04"] }	{ "chsresponse": [{ "family": ["windows"], "global_score": 1.2, "group": "windows 10", "hostname": ["sp-test-laptop"], "ip_address": ["192.168.x.x"], "local_score": 92.21, "os": ["Microsoft Windows 10"], "score": 19.4 },

Sample request	Sample response
	{ "family": ["unix"], "global_score": 1.2, "group": "ubuntu", "hostname": ["sp-test2-desktop"], "ip_address": ["192.168.x.x"], "local_score": 46.17, "os": ["Ubuntu 20.04"], "score": 10.19 }] }

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.
- No Provision.

Get Org With Accounts Hygiene Score Trend

This GET method fetches the scores for the last 30 days if available. If the organization doesn't contain the scores from the last 30 days, it will return them from when they were first calculated until the API was called. If the organization score is not calculated on some of the last 30 days, the most recently calculated scores according to those missing days will be displayed on those days.

Method Name: **getOrgWithAccountsHygieneScoreTrend**

Method Type: GET

Mandatory Parameters: account_name

URL: https://saner.secpod.com/CHScanner/getOrgWithAccountsHygieneScoreTrend

Sample request	Sample response
{saner.secpod.com address}/CHScanner/getOrgWithAcountsHygieneScoreTrend?organization=organization name	{ "chsresponse": [{ "account_name": "Integration Demo", "trending_score": [{ "account_id": "spxxxxxxxxxxx5", "account_name": "Integration Demo", "account_score": "", "anomaly_score": "", "best_possible_score": 99.6, "ccss_score": "", "cvss_score": "", "date": "2025-02-26", "missing_patch_score": "",

Sample request	Sample response
	"no_of_devices": "", "rp_score": "", "saner_remediable_score": 99.6, "worst_possible_score": 99.6 }] }, { "account_name": "Feature Demo", "trending_score": [{ "account_id": "sxxxxxxxxxxxxx6", "account_name": "Feature Demo", "account_score": "", "anomaly_score": "", "best_possible_score": 0, "ccss_score": "", "cvss_score": "", "date": "2025-02-26", "missing_patch_score": "", "no_of_devices": "", "rp_score": "", "saner_remediable_score": 0, "worst_possible_score": 0 }] }, { "account_name": "Demo Account", "trending_score": [{ "account_id": "sxxxxxxxxxxxxxm", "account_name": "Demo Account", "account_score": 22.52, "anomaly_score": 8.74, "best_possible_score": "", "ccss_score": 0.26, "cvss_score": 30.21, "date": "2025-01-26", "missing_patch_score": 21.91, "no_of_devices": 4, "rp_score": 16.36, "saner_remediable_score": 53.86, "worst_possible_score": "" }] }] }

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.
- 403 Forbidden

Get Trending CHScore

This GET method fetches the scores for the last 30 days if available. If the account doesn't contain the last 30 days scores, then it will return the scores from the time they were first calculated until the time the API was called. If account scores are not calculated on some of the last 30 days, the most recently calculated scores according to those missing days will be displayed on those days.

Method Name: getTrendingCHScore

Method Type: GET

Mandatory Parameters: account_name

URL: https://saner.secpod.com/CHScanner/getTrendingCHScore

Sample request	Sample response
{saner.secpod.com address}/CHScanner/getTrendingCHScore?account_name=account name	{ "chsresponse": [{ "account_id": "sp1XXXXXX", "account_name": "Account_1", "account_score": 57.37, "anomaly_score": 12.24, "ccss_score": 13.31, "cvss_score": 12.34, "date": "2023-03-23", "missing_patch_score": 4.74, "no_of_devices": 94 }, { "account_id": "spXXXXXX", "account_name": "Account_1", "account_score": 86.33, "anomaly_score": 0.8, "ccss_score": 0.0, "cvss_score": 0.0, "date": "2023-03-24", "missing_patch_score": 12.87, "no_of_devices": 94 }....] }

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.
- No Provision.

Get CHS Frequency Distribution

This method returns the number of devices per score interval.

Method Name: getCHSFrequencyDistribution

Mandatory Parameters: account_name

Method Type: POST

URL: https://saner.secpod.com/CHScanner/getCHSFrequencyDistribution

Sample request	Sample response
{ "account_name": "{{account_name}}", "family": ["windows", "unix", "macos"], "devices": ["sp-test-laptop", "sp- test2-desktop"], "group": ["windows 10", "ubuntu"], "os": ["Microsoft Windows 10", "Ubuntu 20.04"], "score_interval": 20 }	{ "chsresponse": [{ "host_count": 2, "hostname": ["sp-test-laptop", "sp-test2-desktop"], "score": "0-20" }, { "host_count": 0, "hostname": [], "score": "20-40" }, { "host_count": 0, "hostname": [], "score": "40-60" }, { "host_count": 0, "hostname": [], "score": "60-80" }, { "host_count": 0, "hostname": [], "score": "80-100" }] }

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.
- No Provision.

Get CHS Report For Top Contributors (PA)

This method returns the top five contributors of PA with hosts, name, and weightage.

Method Name: getTopCHSAttributesOfPA

Method Type: POST

Mandatory Parameters: account_name

URL: https://saner.secpod.com/CHScanner/getTopCHSAttributesOfPA

Sample request	Sample response
{ "account_name": "<Account Name>" }	{ "chsresponse": [{ "hosts": ["sp-test-laptop"],

Sample request	Sample response
	"host_count": 1, "name": "PA-2022-1002", "high": 0, "low": 5104, "medium": 0, "weightage": 5104.0 }, { "hosts": ["sp-test-laptop"], "host_count": 1 "name": "PA-2022-1002", "high": 0, "low": 5104, "medium": 0, "weightage": 5104.0 }, ...] }

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.
- No Provision.

CHS Report For Top Contributors (VM)

This method returns the top five contributors of VM with hosts, name, severity, and weightage.

Method Name: **getTopCHSAttributesOfVM**

Method Type: POST

Mandatory Parameters: account_name

URL: https://saner.secpod.com/CHScanner/getTopCHSAttributesOfVM

Sample request	Sample response
{ "account_name": "<Account Name>" }	{ "chsresponse": [{ "hosts": ["sp-test-laptop"], "host_count": 1,"name": "CVE-2022-38045", "severity": "critical", "weightage": 410.42 }, { "hosts": ["sp-test-laptop"], "host_count": 1 "name": "CVE-2022-38045", "severity": "critical", "weightage": 410.42 }, ...

Sample request	Sample response
	] }

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.
- No Provision.

CHS Report For Top Contributors (CM)

This method returns top five contributors of CM with hosts, name, severity, and weightage.

Method Name: **getTopCHSAttributesOfCM**

Method Type: POST

Mandatory Parameters: account_name

URL: https://saner.secpod.com/CHScanner/getTopCHSAttributesOfCM

Sample request	Sample response
{ "account_name": "<Account Name>" }	{ "chsresponse": [{ "hosts": ["sp-test-laptop"], "host_count": 1,"name": "CCE-41482-1", "severity": "critical", "weightage": 297.0 }, { "hosts": ["sp-test-laptop"], "host_count": 1 "name": "CCE-41482-1", "severity": "critical", "weightage": 297.0 }, ...] }

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.
- No Provision.

CHS Report For Top Contributors (PM)

This method returns top five contributors of PM with hosts, name, severity, and weightage.

Method Name: getTopCHSAttributesOfPM

Method Type: POST

Mandatory Parameters: account_name

URL: https://saner.secpod.com/CHScanner/getTopCHSAttributesOfPM

Sample request	Sample response
{ "account_name": "<Account Name>" }	{ "chsresponse": [{ "hosts": ["sp-test-laptop"], "host_count": 1,"name": "https://logging.apache.org/log4j/2.x/download.html ", "severity": "critical", "weightage": 80 }, { "hosts": ["sp-test-laptop"], "host_count": 1 "name": "https://logging.apache.org/log4j/2.x/download.html ", "severity": "critical", "weightage": 80 }, ...] }

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.
- No Provision.

Get Org CHScore By Family

This method returns the CHS Score of the Organization based on the family.

Method Name: getOrgCHScoreByFamily

Method Type: POST

Mandatory Parameters: account_name

URL: https://saner.secpod.com/CHScanner/getOrgCHScoreByFamily

Sample request	Sample response
{ "organization": "Test Org" }	{ "chsresponse": [{ "macos": 54.895514173424964,

Sample request	Sample response
	"organization": "Demo", "others": 56.72493327423742, "unix": 38.2792968453746, "windows": 54.677358812713415 }] }

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.

Get Org CHScore By Group

This method returns the CHS Score of the Organization based on the group.

Method Name: getOrgCHScoreByGroup

Method Type: POST

Mandatory Parameters: account_name

URL: https://saner.secpod.com/CHScanner/getOrgCHScoreByGroup

Sample request	Sample response
{ "organization": "Test Org" }	{ "chsresponse": [{ "": 54.677358812713436, "alpine": 45.737645027782236, "centos": 7.128198198198198, "debian": 53.51922647451805, "general purpose": 54.17159225468311, "linuxmint": 53.927092055242774, "mac os": 29.25830731450623, "organization": "Demo", "phone": 54.06263862781957, "ubuntu": 48.74996064570943, "windows 10": 53.017379263493254, "windows 11": 54.73505423280425, "windows 7": 47.58666399438158, "windows server 2016": 40.66070909692875 }] }

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.

Get Org CHS Score By OS

This method returns the CHS Score of the Organization based on the operating system.

Method Name: getOrgCHScoreByOS

Method Type: POST

Mandatory Parameters: organization

URL: https://saner.secpod.com/CHScanner/getOrgCHScoreByOs

Sample request	Sample response
{ "organization": "Test Org" }	{ "chsresponse": [{ "Alpine Linux 3.12": 40.27396377960813, "Android 7.1.2 (Linux 3.10)": 53.133138075313816, "Apple Mac OS 11.6": 55.06022940839106, "Apple Mac OS 12.5": 51.92737414075287, "Apple Mac OS 12.6": 52.37657475728156, "Apple Mac OS X 10.15": 54.63779180554611, "CentOS 6.7": 51.68926502290488, "CentOS 7.9": 52.5850050161632, "Debian": 54.677358812713436, "Linux 2.6.18": 53.28748308609026, "Linux 2.6.32 - 3.10": 56.056435056669535, "Linux 4.15 - 5.6": 56.13844386290213, "LinuxMint 21": 52.82595711329865, "Microsoft Windows 10": 49.8204937378388, "Microsoft Windows 11": 55.42747154543838, "Microsoft Windows 7 Service Pack 1": 54.41023436538943, "Microsoft Windows Server 2016": 45.57010348668349, "Microsoft Windows Vista Home Premium SP1": 53.053464753587036, "Ubuntu 14.04": 51.72590217755444, "Ubuntu 16.04": 43.233945133664875, "Ubuntu 18.04": 17.105252660688844, "Ubuntu 20.04": 52.79542173490949, "Ubuntu 21.04": 53.32768410195721, "Ubuntu 22.04": 52.77690442864341, "organization": "Demo" }] }

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.

Get Org Hygiene Score

This GET method returns the organization's score and score details of the accounts that are calculated and present in the given organization.

Method Name: getOrgHygieneScore

Method Type: GET

Mandatory Parameters: organization

URL: https://saner.secpod.com/CHScanner/getOrgHygieneScore

Sample request	Sample response
{saner.secpod.com address}/CHScanner/getOrgHygieneScore?organization=Organization name	{ "chsresponse": [{ "accounts": [{ "account_id": "spxxxxx", "account_name": "Test_Account1", "account_score": 18.15, "anomaly_score": 79.2, "ccss_score": 1.5, "cvss_score": 0.43, "missing_patch_score": 0.72, "no_of_devices": 83, "timestamp": "2023-05-24 05:27:40.450122" }, { "account_id": "spxxxxx", "account_name": "abc", "account_score": "No Info", "anomaly_score": "No Info", "ccss_score": "No Info", "cvss_score": "No Info", "missing_patch_score": "No Info", "no_of_devices": "No Info", "timestamp": "No Info" }, { "account_id": "spxxxxx", "account_name": "Test_Account2", "account_score": "No Info", "anomaly_score": "No Info", "ccss_score": "No Info", "cvss_score": "No Info", "missing_patch_score": "No Info", "no_of_devices": "No Info", "timestamp": "No Info" }], "org_name": "Test_Account", "org_score": 18.15 }] }

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.
- No provision.

Get Top CHS Attributes

This GET method returns the top five anomalies with scores, ccs with scores, cvs with scores, and missing patches with scores.

Method Name: getTopCHSAttributes

Method Type: GET

Mandatory Parameters: account_name

URL: https://saner.secpod.com/CHScanner/getTopCHSAttributes

Sample request	Sample response
{saner.secpod.com address}/CHScanner/getTopCHSAttributes?account_name=account name	{ "anomaly": [{ "hosts": ["sp-test-laptop"], "name": "PA-2022-1073", "severity": { "high": 90062, "low": 79, "medium": 0 }, "weightage": 360327.0 }], "ccss": [{ "description": { "scapMetadatas": { "metadata": [{ "cce": { "description": { "description_data": [{ "value": "Disable: 'Prevent installation of removable devices'\n\nThis policy setting allows you to prevent Windows from installing removable devices. A device is considered removable when the driver for the device to which it is connected indicates that the device is removable. For example, a Universal Serial Bus (USB) device is reported to be removable by the drivers for the USB hub to which the device is connected. This policy setting takes precedence over any other policy setting that allows Windows to install a device.\n If you enable this policy setting, Windows is prevented from installing removable devices and existing removable devices cannot have their drivers updated. If you enable this policy setting on a remote desktop server, the policy setting affects redirection of removable devices from a remote desktop client to the remote desktop server.\n If you disable or do not configure this policy setting, Windows can install and update device drivers for removable devices as allowed or prevented by other policy settings.\n\n Counter Measure:\n Configure this setting depending on your organization's requirements.\n\n Potential Impact:\n Users are unable to install device drivers for new removable devices and are unable to update device drivers for existing removable devices." }]

Sample request	Sample response
	} }, "created_date": "2016-09-23", "modified_date": "2023-10-10", "parameters": [], "references": [{ "reference": "oval:org.secpod.oval:def:35364", "resource_id": "SCAP Repo OVAL Definition" }], "technical_mechanism": "(1) GPO: Computer Configuration\\Administrative Templates\\System\\Removable Storage Access\\All Removable Storage classes: Deny all access\n\n(2) REG: HKEY_LOCAL_MACHINE\\SOFTWARE\\Policies\\Microsoft\\Windows\\RemovableStorageDevices!Deny_All" }] } }, "detection_date": "2018-12-20", "hosts": ["sp-test-laptop"], "name": "CCE-43960-4", "released_date": "2016-09-23", "severity": "high", "title": "Prevent installation of removable devices", "weightage": 311.5 }], "cvss": [{ "description": { "scapMetadatas": { "metadata": [{ "created_date": "2022-10-31", "cve": { "description": { "description_data": [{ "value": "Type confusion in V8 in Google Chrome prior to 107.0.5304.87 allowed a remote attacker to potentially exploit heap corruption via a crafted HTML page. (Chromium security severity: High)" }] }, "references": { "references": { "reference_data": [{ "name": "https://chromereleases.googleblog.com/2022/10/stable-channel-update-for-desktop_27.html", "url": "https://chromereleases.googleblog.com/2022/10/stable-channel-update-for-desktop_27.html" }, { "name": "https://crbug.com/1378239", "url": "https://crbug.com/1378239" }] } } }, "impact": { "baseMetricV2": { "cvssV2": { "baseScore": 0 },

Sample request	Sample response
	"impactScore": 0 }, "baseMetricV3": { "cvssV3": { "attackComplexity": "LOW", "attackVector": "NETWORK", "availabilityImpact": "HIGH", "baseScore": 8.8, "confidentialityImpact": "HIGH", "integrityImpact": "HIGH", "privilegesRequired": "NONE", "scope": "UNCHANGED", "userInteraction": "REQUIRED" }, "exploitabilityScore": 2.8, "impactScore": "5.9" } }, "modified_date": "2022-11-18" }] } }, "detection_date": "2022-10-29", "hosts": ["sp-test1-laptop"], "name": "CVE-2022-3723", "released_date": "2022-10-31", "severity": "critical", "title": "Type confusion vulnerability in V8 in Google Chrome via unspecified vectors - CVE-2022-3723", "weightage": 917.06 }], "missing_patch": [{ "detection_date": "2022-10-29 08:10:28 AM UTC", "hosts": ["sp-test2-laptop"], "name": "google-chrome-107.0.5304.107-x64.exe", "reboot_status": "FALSE", "release_date": "", "severity": "critical", "vendor": "google", "weightage": 160 }] }

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.
- No provision.

Get CHS Weightage

This method returns the weights of anomaly, compliance, missing patches and vulnerabilities in calculation of cyber hygiene score.

Method Name: getCHSWeightage

Method Type: GET

Mandatory Parameters: account_name

URL: https://saner.secpod.com/CHScanner/getCHSWeightage

Sample request	Sample response
{saner.secpod.com address}/CHScanner/getCHSWeightage?account_name=Account name	{ "service_provision": { "CM": true / false, "EDR": true / false, "PM": true / false, "VM": true / false }, "timestamp": "", "weightage": { "Anomaly": , "Compliance": , "Missing_Patches": , "Vulnerability": } }

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.
- No provision.

Update CHS Weightage

This method is used to update the weights of anomaly, compliance, missing patches, vulnerabilities and starts a CHS Scan.

Method Name: updateCHSWeightage

Method Type: POST

Mandatory Parameters: account_name, user_id, and weightage

URL: https://saner.secpod.com/CHScanner/updateCHSWeightage

Sample request	Sample response
{ "account_name": "Test Account", "user_id": "test@secpod.com", "weightage": { "Anomaly": 20, "Compliance": 20, "Missing_Patches": 20, "Risk_Prioritization": 20, "Vulnerability": 20 }	{ "message": "Successful" }

Sample request	Sample response
}	

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.
- No provision.
- User doesn't have access.
- Sum of weights should be 100.

Get Trending Org Hygiene Score

This GET method fetches the score of the last 30 days of the Organization if they are available and follows the below-mentioned three conditions.

Fetches the scores of the last 30 days of the organization if they are available.

→ If the organization doesn't contain the last 30 days scores, then it will return the scores from the time they were first calculated until the time the API was called.

→ Details takes string values. Default value of details is "false". If it's "true", the response also contains the available scores of accounts under that organization for last 30 days, which were used to calculate the trending organization scores.

→ If organization scores are not calculated on some of the last 30 days, the most recently calculated scores according to those missing days will be displayed on those days.

Method Name: getTrendingOrgHygieneScore

Method Type: GET

Mandatory Parameters: organization

URL: https://saner.secpod.com/CHScanner/getTrendingOrgHygieneScore

Sample request	Sample response
{server ip ddress}/CHScanner/getTrendingOrgHygieneScore?organization=Organization name	If details :True { "chsresponse": [{ "date": "2023-10-07", "org_name": "Saner APIs", "org_score": 13.73 }, { "date": "2023-10-08", "org_name": "Saner APIs", "org_score": 10.96 }, { "date": "2023-10-09", "org_name": "Saner APIs",

Sample request	Sample response
	<pre>"org_score": 10.96 }, { "date": "2023-10-10", "org_name": "Saner APIs", "org_score": 10.96 }, { "date": "2023-10-11", "org_name": "Saner APIs", "org_score": 10.96 }] } _____ If details :False { "chsresponse": [{ "date": "2023-04-24", "org_name": "Test Account", "org_score": 18.18 }, ..] }</pre>

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.

Get CHScan Status

This method returns the CHS scan status of an Account.

Method Name: **getCHScanStatus**

Method Type: GET

Mandatory Parameters: account_name

URL: https://saner.secpod.com/CHScanner/getCHScanStatus

Sample request	Sample response
<pre>{saner.secpod.com address}/CHScanner/getCHScanStatus?account_name=Account name</pre>	<pre>The API response returns the scanning status of an Account. { "message": "Account Ready for Scan" } OR</pre>

Sample request	Sample response
	{ "message": "Scan Ongoing" }

Possible Error Cases

- Bad Request.
- Account not found or expired.
- Schedule the CHS Scan.
- No provision.

Update Org CHS Weightage

This method updates the weights of accounts under the given organization. Scans the accounts after updating.

Method Name: updateOrgCHSWeightage

Method Type: POST

Mandatory Parameters: organization, weightage, and user_id

URL: https://saner.secpod.com/CHScanner/updateOrgCHSWeightage

Sample request	Sample response
{ "organization": "Test Org", "weightage": { "Risk_Prioritization": 20, "Vulnerability": 20, "Missing_Patches": 20, "Compliance": 20, "Anomaly": 20 }, "user_id": "test@secpod.com", "account_names": ["Test Account"] }	{ "message":"Successful" }

Possible Error Cases

- Bad Request.
- Sum of weights should be 100.
- Account not found or expired.
- Access denied.

Asset Exposure

Saner Asset Exposure enables organizations to track, monitor, and manage software and hardware assets across their IT environment. It provides visibility into asset inventory and exposure, helping identify vulnerable, risky, or non-compliant applications. This section outlines the Saner Asset Exposure REST APIs and their usage.

Add Whitelisted Assets

Adds one or more applications (assets) to the whitelist.
Whitelisted assets represent approved applications that are permitted within the environment. Applications added to the whitelist override any existing blacklisted application and are considered as trusted in the Asset Exposure (AE) dashboard.

Method Name: addWhiteListedAssets

Method Type: POST

Mandatory Parameters: account, organization, assets, applicationname, version

URL: https://saner.secpod.com/sanerapi/v1.0/AE/assetlist/addwhitelistedassets

Sample Request

```
{
  "request": {
    "method": "addWhiteListedAssets",
     "parameters": {
       "parameterset": [{
         "parameter": [{
           "key": "account",
           "value": "Saner-Account"
                        },
          {
            "key": "organization",
            "value": "Saner-Org"
                        },
          {
            "key": "assets",
            "value": [{
             "applicationname": "boost-
atomic",
             "version": "1.66.0",
             "publisher":""

                                  }]
                        }]
                   }]
         }
     }
}
```

Sample Response

In case of success:

```
{
  "status":"success",
  "reason":"Assets added successfully"
}
```

In case of failure:

```
{
  "status":"fail",
  "reason":"Account not found"
}
```

Add Blacklisted Assets

Adds one or more application (assets) to the blacklist.

Blacklisted assets represent applications that are restricted, unauthorized, or pose a security risk within the environment. When a blacklisted application is detected on any endpoint, it is flagged in the Asset Exposure (AE) dashboard, enabling administrators to identify violations and take appropriate actions.

Method Name: addBlackListedAssets

Method Type: POST

Mandatory Parameters: account, organization, assets, applicationname, version

URL: https://saner.secpod.com/sanerapi/v1.0/AE/assetlist/addblacklistedassets

Sample Request	Sample Response
<pre>{ "request": { "method": "addBlackListedAssets", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "Saner-Account" }, { "key": "organization", "value": "Saner-Org" }, { "key": "assets", "value": [{ "applicationname": "boost-atomic", "version": "1.66.0", "publisher":"" }] }] }] } } }</pre>	In case of success: <pre>{ "status":"success", "reason":"Assets added successfully" }</pre> In case of failure: <pre>{ "status":"fail", "reason":"Failed to update the asset as it already exists in the selected category." }</pre>

Posture Anomaly

Saner Posture Anomaly identifies the anomalies in your Organization's IT infrastructure. This ranges from identifying devices that are misconfigured, have a unique posture, and are configured completely differently when compared to other devices.This section provides insights into the Saner Posture Anomaly APIs and their usage.

Initiate Posture Anomaly Scan

This method is used to start the PA scan for an existing account.

Method Name: initiatePostureAnomalyScan

Method Type: POST

Mandatory Parameters: organization and account_name

URL: https://saner.secpod.com/AncorWebService/perform

Sample request	Sample response
<pre>{ "request": { "method": "initiatePostureAnomalyScan", "parameters": {</pre>	<pre>{ "response": { "method": "initiatePostureAnomalyScan", "results": {</pre>

Sample request	Sample response
<pre>"parameterset": [{ "parameter": [{ "key": "organization", "value": "Test Org" }, { "key": "account_name", "value": "Test Account" }]}]}}}</pre>	<pre>"result": [{ "status": "success", "reason": "" }] } } }</pre>

Possible Error Cases

- Scan is already running.
- Invalid account name.

Get Posture Anomaly Scanner Config

This method is used to fetch existing scan configurations or the PA scan schedule that has already been set for an account.

Method Name: **getPostureAnomalyScannerConfig**

Method Type: POST

Mandatory Parameters: organization and account_name

URL: https://saner.secpod.com/AncorWebService/perform

Sample request	Sample response
<pre>{ "request": { "method": "getPostureAnomalyScannerConfig", "parameters": { "parameterset": [{ "parameter": [{ "key": "organization", "value": "Test Org" }, { "key": "account_name", "value": "Test Account" }]}]}}}</pre>	<pre>{ "starttime": "10:00:AM", "month-of-year": "12", "day-of-month": "", "week-of-month": "1", "day-of-week": "3,6", "schedule": "monthly" }</pre>

Possible Error Cases

- Invalid account name.

Add Posture Anomaly Scanner Config

This method is used to add a scheduled scan to perform a PA scan for an existing account.

Method Name: addPostureAnomalyScannerConfig

Method Type: POST

Mandatory Parameters: organization and account_name

URL: https://saner.secpod.com/AncorWebService/perform

Sample request	Sample response
<pre>{ "request": { "method": "addPostureAnomalyScannerConfig", "parameters": { "parameterset": [{"parameter": [{"key": "organization", "value": "Test Org" },{ "key": "account_name", "value": "Test Account" },{"key": "schedule", "value": "monthly" },{"key": "starttime", "value": "10:00:AM" },{"key": "monthoftheyear", "value": "12" },{"key": "weekofthemonth", "value": "1" },{"key": "dayoftheweek", "value": "3,6" }]}]}}</pre>	<pre>{ "response": { "method": "addPostureAnomalyScannerConfig", "results": { "result": [{ "key": "spxxxxx", "status": "success", "reason": "" }]}}}</pre>

Possible Error Cases

- Invalid account name.
- Incorrect date.
- Incorrect month
- Incorrect time.

Delete Posture Anomaly Scanner Config

This method is used to remove the existing PA scan schedule set for the account.

Method Name: deletePostureAnomalyScannerConfig

Method Type: POST

Mandatory Parameters: organization and account_name

URL: https://saner.secpod.com/AncorWebService/perform

Sample request	Sample response
<pre>{ "request": { "method": "deletePostureAnomalyScannerConfig", "parameters": { "parameterset": [{ "parameter": [{ "key": "organization", "value": "Test Org" }, { "key": "account_name", "value": "Test Account" }]}]}}}</pre>	<pre>{ "response": { "method": "deletePostureAnomalyScannerConfig", "results": { "result": [{ "status": "success", "reason": "" }]}}}</pre>

Possible Error Cases

- Invalid account name.

Get Status

This method is used to get the Posture Anomaly scan status for a given account.

Method Name: getStatus

Method Type: POST

Mandatory Parameters: account name

URL: https://saner.secpod.com/PAScanner/getStatus

Sample request	Sample response
<pre>{ "account_name": "{{account name}}" }</pre>	<pre>When PA Scan is ongoing: { "message": "account under update" } When PA Scan is completed: { "message": "account update finished" }</pre>

Possible Error Cases

- Account not found or expired.
- PA not enabled.
- Data not found.
- Wrong PA.

Get Posture Anomaly

This method is used to get the Posture Anomaly data for a given account.

Method Name: getPostureAnomaly

Method Type: POST

Mandatory Parameters: account name and PA-ID

URL: https://saner.secpod.com/PAScanner/getPostureAnomaly

Sample request	Sample response
<pre>{ "account_name": "{{account name}}", "PA-ID": "PA-2022-1001" }</pre>	<pre>{ "ID": "PA-2022-1001", "category": "System", "data": { "linux": [{ "anomaly": "anomaly", "confidence": "low", "details": { "devices": [{ "device-type": "end-point", "family": "unix", "group": "centos", "hostname": "qa-centos-6-x86", "ip": "192.168.1.188", "mac": "02-59-22-9B-E8-82", "os": "CentOS 6.7" }] }, "frequency": 1, "ver": "2.6.32-573.el6.i686" }], "mac": [] }, "desc": "Identifying anomalous Kernel versions across the devices. This process involves data collection, transformation, clustering, tuning and alerting based on lower bound.", "title": "Atypical Kernel version found", "type": "outlier-based" }</pre>

Possible Error Cases

- Account not found or expired.
- PA not enabled.
- Data not found.
- Wrong PA-ID

Get Configuration

This method is used to get PA configured entries for a given account.

Method Name: getConfiguration

Method Type: POST

Mandatory Parameters: account name

URL: https://saner.secpod.com/PAScanner/getConfiguration

Sample request

```
{
    "account_name": "{{account name}}"
}
```

Sample response


```
{
    "configure_items": [

        {
            "configure": [

"@%SystemRoot%\\system32\\drivers\\todo.sys,
-101;NDIS Proxy"
            ],
            "devices": "",
            "family": "windows",
            "group": "",
            "id": "PA-2022-1069",
            "key": "service_displayname",
            "reference": "All Services -
Windows",
            "type": "serviceinfo"
        },
        {
            "configure": [
                "abrt-vmcore.service"
            ],
            "devices": "",
            "family": "unix",
            "group": "",
            "id": "PA-2022-1069",
            "key": "service_displayname",
            "reference": "All Services -
Linux",
            "type": "serviceinfo"
        },
        {
            "configure": [

"com.apple.accessibility.mediaaccessibilityd
",
                "test"
            ],
            "devices": "",
            "family": "macos",
            "group": "",
            "id": "PA-2022-1069",
            "key": "service_displayname",
            "reference": "All Services -
Mac",
            "type": "serviceinfo"
        },
        {
            "configure": [
                "http"
            ],
            "devices": "",
            "family": "others",
            "group": "",
            "id": "PA-2022-1069",
            "key": "service_displayname",
            "reference": "All Services -
Others",
            "type": "serviceinfo"
        }
```

Sample request	Sample response
	] }

Possible Error Cases

- Account not found or expired.
- PA not enabled.
- Data not found.
- Wrong PA.

Get Configuration Status

This method is used to check if PA-IDs are configured for an account.

Method Name: **getConfigurationStatus**

Method Type: POST

Mandatory Parameters: account name

URL: https://saner.secpod.com/PAScanner/getConfigurationStatus

Sample request	Sample response
{ "account_name": "{{account name}}" }	{ "PA-2022-1068": ["windows"], "PA-2022-1069": ["unix"], "PA-2022-1070": [], "PA-2022-1071": [], "PA-2022-1072": [], "PA-2022-1073": ["unix"] }

Possible Error Cases

- Account not found or expired.
- PA not enabled.
- Data not found.
- Wrong PA.

Get Whitelist

This method is used to fetch fully whitelisted PA-ID's for a given account.

Method Name: **getWhitelist**

Method Type: POST

Mandatory Parameters: account_id

URL: https://saner.secpod.com/PAScanner/getWhitelist

Sample request	Sample response
<pre>{ "account_name": "{{account name}}" }</pre>	<pre>{ "whitelist_items": [{ "type": "detection", "whitelist": ["PA-2022-1001"] }] }</pre>

Possible Error Cases

- Wrong account.
- PA not enabled.
- Data not found.
- Wrong PA.

Post Whitelist

This method is used to update whitelist PA-IDs for a given account. However, in scenarios where you want to retain the existing whitelisted PA IDs and add new PA IDs to the existing whitelist, we recommend you take the response from the *Get Whitelist* API, add the new PA IDs, and pass it as a request to the *Post Whitelist* API.

Method Name: postWhitelist
Method Type: POST
Mandatory Parameters: account name

URL: https://saner.secpod.com/PAScanner/postWhitelist

Sample request	Sample response
<pre>{ "account_name": "{{account name}}", "whitelist": { "whitelist_items": [{ "type": "detection", "whitelist": ["PA-2022-1001", "PA-2022-1002", "PA-2022-1003", "PA-2022-1004", "PA-2022-1005"] }] }, "scan": "false" }</pre>	<pre>{ "message": "Successful" }</pre>

Possible Error Cases

- Wrong account.
- PA not enabled.
- User doesn't have write access.

- Data not found.

Get All Configuration

This method is used to get all the configurable entries for a given account.

Method Name: getAllConfiguration

Method Type: POST

Mandatory Parameters: account name and PA-ID

URL: https://saner.secpod.com/PAScanner/getAllConfiguration

Sample request	Sample response
<pre>{ "account_name": "{{account name}}", "PA-ID": "PA-2022-1068" }</pre>	<pre>{ "ID": "PA-2022-1072", "category": "Devices", "data": [{ "confidence": "High", "details": [{ "base_board_serial_number": "", "bios_serial_number": "", "boot_device": "\\Device\\HarddiskVolume1", "cache_size": "0", "cpu": "Intel(R) Xeon(R) CPU X5650 @ 2.67GHz", "cpu_arch": "x64", "cpu_cores": "2", "cpu_usage": "2", "disk_description": "Disk drive", "disk_drive_serial_number": "QM00001", "disk_name": "QEMU HARDDISK ATA Device", "disk_size": "107372805120", "disk_type": "Fixed hard disk media", "name": "computer_item", "ntp_status": "True", "nw_received_bytes": "0", "nw_total_bytes": "0", "nw_transmitted_bytes": "0", "operating_system_serial_number": "00331-10000-00001-AA558", "os_arch": "64-bit", "os_name": "Microsoft Windows 10", "ram": "4244090880", "ram_free": "2485702656", "ram_usage": "41", "ram_used": "1758388224", "sys_manufacturer": "QEMU", "sys_type": "Other", "sys_uuid": "BDE441FB-C30A-4D3A-A148- 2209F9CE711E", "system_name": "DESKTOP-JDN034T",</pre>

Sample request	Sample response
	"system_product_name": "Standard PC (i440FX + PIIX, 1996)", "system_product_version": "pc-i440fx-5.1", "system_uptime": "0 Weeks, 6 Days, 1 Hour, 14 Minutes, 11 Seconds", "timezone_diff": "5.5", "timezone_name": "India Standard Time", "volume_name": "\\\\.\\PHYSICALDRIVE0" }], "device": { "device-type": "end-point", "family": "windows", "group": "windows 10", "hostname": "desktop-jdn034t", "ip": "192.168.1.88", "mac": "86-25-F3-F7-4B-FF", "os": "Microsoft Windows 10" } }], "desc": "Detecting unwanted devices, this detection mechanism needs user configuration and inputs.", "message": "Not Configured", "title": "Unwanted Devices", "type": "query" }

Possible Error Cases

- Wrong Account
- PA not enabled.
- Data not found.
- Wrong PA-ID.

Vulnerability Management

Saner Vulnerability Management enables continuous detection, assessment, and tracking of vulnerabilities across your organization's IT assets. It helps administrators understand vulnerability severity and exposure across endpoints, servers, and applications. This section provides an overview of the Saner Vulnerability Management REST APIs and their usage.

Exclude a Vulnerability

Excludes a vulnerability from remediation and visibility based on the specified scope. This API allows administrators to create exclusion policies for vulnerabilities (CVEs) when visibility or remediation is not required, not applicable, risk accepted or temporarily deferred. Exclusions can be applied at the account, group, or device level. Exclusions are enforced for the defined duration and automatically removed once the exclusion period expires.

Method Name: excludevulnerability

Method Type: POST

Mandatory Parameters: account_name, policy_name, item_type, items, scopeType, selectedScopeItems, module_type

Sample Request

```
{
  "request": {
    "method": "excludevulnerability",
     "parameters": {
       "parameterset": [{
         "parameter": [{
           "key":"account_name",
            "value":"Saner-Account"
                    },
           {
            "key": "policy_name",
            "value": " policy001"
                    },
           {
            "key": "item_type",
            "value": "CVE"
                    },
           {
            "key": "items",
            "values": ["CVE-2023-
5167","CVE-2022-350"]
                    },
           {
```

Sample Response

In case of success:

```
{
    "response": {
        "method": "excludevulnerability",
        "results": {
            "result": [
                {
                    "key": " policy001",
                    "status": "SUCCESS",
                    "reason": "Success!"
                }]
        }
    }
}
```

In case of failure:

```
{
    "response": {
        "method": "excludevulnerability",
        "results": {
            "result": [
```

<table>
<tr>
<td>

```
            "key": "days",
            "value": "1"
                    },
           {
            "key": "reason",
            "value": "RISK_ACCEPTED"
                    },
           {
            "key": "desc",
            "value": ""
                    },
           {
             "key": "comments",
               "value": ""
                    },
            {
             "key": "scopeType",
             "value": "group"
                    },
            {
             "key": "selectedScopeItems",
             "values": ["debian12"]
                    },
            {
             "key": "module_type",
             "value": "VM"
                    }]
            }]
        }
    }
}
```

</td>
<td>

```
                {
                    "key": " policy001",
                    "status": "FAIL",
                    "reason": "Duplicate
exclude policy name"
                }]
        }
    }
}
```

</td>
</tr>
</table>

Compliance Management

Saner Patch Management tool supports various compliance benchmarks such as NIST, PCI, and HIPAA. You can perform tasks such as Adding, provisioning and deleting a benchmark. Also, using Saner Patch Management APIs, you can retrieve misconfigurations and perform remediations.

This section provides insights into the Saner Compliance Management APIs and their usage.

Add Benchmark

This method is used to assign compliance benchmarks to one or more accounts.

Method Name: addBenchmark

Method Type: POST

Mandatory Parameters: account_name, target_group, and benchmarkname.

Sample request	Sample response
<pre>{ "request": { "method": "addbenchmark", "parameters": {</pre>	<pre>In case of success: { "response": { "method": "addbenchmark",</pre>

<table>
<tr>
<td>

```
      "parameterset": [{
        "parameter": [{
          "key": "account name",
          "value": "testaccount1"
        },{
          "key": "benchmarkname",
          "value": "ComplianceWin7"
        },{
          "key":
"target_account_name",
          "value": "ALL"
        },{
          "key": "target_group",
          "value": "windows 7"
        }]
      }]
    }
  }
}
```

</td>
<td>

```
    "results": {
      "result": [{
        "key": "Failed to add ComplianceWin7 for
testaccount1",
        "status": "FAIL",
        "reason": "User/Account need subscription to
perform the operation."
      },{
        "key": "Added ComplianceWin7 for
testaccount2",
        "status": "SUCCESS",
        "reason": ""
      }]
    }
  }
}
```

</td>
</tr>
</table>

Possible Error Cases

- Benchmark already exist in the target account.
- Benchmark not found in the source account.
- Input validation error.
- Service-disabled error.

Delete Benchmark

This method allows you to delete an existing benchmark. However, you can't delete a benchmark if it is associated with an administrator.

Method Name: deleteBenchmark

Method Type: POST

Mandatory Parameters: account_name and benchmarkname

<table>
<tr><th>Sample request</th><th>Sample response</th></tr>
<tr>
<td>

```
{
  "request": {
    "method": "deletebenchmark",
    "parameters": {
      "parameterset": [{
        "parameter": [{
          "key": "account_name",
          "value": " secpodaccounttest1"
        },{
          "key": "benchmarkname",
          "value": "BenchmarkWindows10"
        }]
      }]
    }
  }
}
```

</td>
<td>

In case of success:

```
{
  "response": {
    "method": "deletebenchmark",
    "results": {
      "result": [{
        "key": "",
        "status": "SUCCESS",
        "reason": " Benchmark Deleted."
      }]
    }
  }
}
```

</td>
</tr>
</table>

Possible Error Cases

- Benchmark not found.

- Benchmark provision does not exist.
- Input validation error.
- Service is disabled.

Get All Benchmark Provision

This method provides the benchmark name, source account name, target account name, and target organization name.

Method Name: getAllBenchmarkProvisions

Method Type: POST

Mandatory Parameters: No mandatory parameters

Sample request	Sample response
{ "request": { "method": "getAllBenchmarkProvisions" } }	In case of success: { "benchmarkProvisions": [{ "benchmarkname": "ubuntu18", "source_account_name": "Test Account", "target_account_name": "Test Account2", "target_organization": "Test" }, { "benchmarkname": "rhel9", "source_account_name": "Test Account", "target_account_name": "Test Account2", "target_organization": "Test" }, { "benchmarkname": "centos", "source_account_name": "Test3", "target_account_name": "Test4, New", "target_organization": "SecPod" }] }

Possible Error Cases

- The specified administrator is not found.
- Service is not enabled.
- Input validation error

Provision Benchmark

This method is used to apply benchmark from one source Account to *ALL* Accounts in the Organization. The '*value: ALL'* indicates that benchmark will be available but not applied to all future Accounts created. If you intend to add benchmark to one or more Accounts in the Organization and not *ALL* Accounts, use 'addbenchmark' method.

Method Name: provisionBenchmark

Method Type: POST

Mandatory Parameters: account_name, benchmarkname, target_account_name, and target_organization

Sample request	Sample response
<pre>{ "request": { "method": "provisionbenchmark", "parameters": { "parameterset": [{ "parameter": [{ "key": "account_name", "value": "Test_Account" }, { "key": "benchmarkname", "value": "Test" }, { "key": "target_organization", "value": "Test_Org" }, { "key": "target_account_name", "value": "all" }, { "key": "target_group", "value": "windows 11" }, { "key": "apifilters", "filters": [{ "tags": { "in": { "Owner": ["User_1", "User_2"] }, "not in": { "Category": ["Finance"] } } }] }] }] } } }</pre>	<pre>In case of success: { "response": { "method": "provisionbenchmark", "results": { "result": [{ "key": "Failed to add ComplianceWin10 for testaccount2", "status": "FAIL", "reason": "User/Account need subscription to perform the operation." },{ "key": "Added ComplianceWin10 for testaccount3", "status": "SUCCESS", "reason": "" }] } } }</pre>

Possible Error Cases

- Benchmark already exist in the target account.
- Benchmark not found in the source account.
- Input validation error.
- Service-disabled error.
- Target account not found under organization.
- Target account not found.

- Organization not found.

Update Benchmark Provision

This method can be used to update accounts for an existing compliance provision. The benchmark will be added to the specified accounts; however, the newly added accounts will continue to get existing compliance provision.

Method Name: updateProvisionBenchmark

Method Type: POST

Mandatory Parameters: adminId, benchmarkname, test_organization, and test_account_name

Sample request	Sample response
{ "request": { "method": "updateProvisionBenchmark", "parameters": { "parameterset": [{ "parameter": [{ "key": "adminId", "value": "user@secpod.com" }, { "key": "benchmarkname", "value": "Test" }, { "key": "target_organization", "value": "TempOrg2" }, { "key": "target_account_name", "value": "all" }, { "key": "target_group", "value": "debian" }, { "key": "apifilters", "filters": [{ "tags": { "in": { "Owner": ["User-1"] }, "not in": { "Category": ["Finance"] } } }] }	In case of success: { "response": { "method": "updateprovisionbenchmark", "results": { "result": [{ "key": "Added ComplianceWin10 for testaccount2", "status": "SUCCESS", "reason": "" }] } } }

Sample request	Sample response
] }] } } }	

Possible Error Cases

- Benchmark not found.
- Service not enabled.
- Input validation error.
- Organization not found.
- Target account not found.
- Target account not found under organization.

Delete Benchmark Provision

This method is used to remove benchmarks added by an administrator and the removal will reflect in all existing accounts and future accounts.

Method Name: deleteProvisionBenchmark

Method Type: POST

Mandatory Parameters: adminId, benchmarkname, target_organization

Sample request	Sample response
{ "request": { "method": "deleteprovisionbenchmark", "parameters": { "parameterset": [{ "parameter": [{ "key": "adminId", "value": " secpodusertest1@secpod.com" },{ "key": "benchmarkname", "value": " BenchmarkWindows10" },{ "key": "target_organization", "value": "organization1" }] }] } } }	In case of success: { "response": { "method": "deleteprovisionbenchmark", "results": { "result": [{ "key": "", "status": "SUCCESS", "reason": " Benchmark Provision deleted." }] } } }

Possible Error Cases

- Benchmark not found.
- Administrator not found.
- Service disabled for administrator.
- Input validation error.
- Organization not found.

Get Applicable Misconfiguration Remediation

This method allows you to list all the misconfiguration patches by group, hostname, or a family for an account.

Method Name: getApplicableMisconfigurationRemediation

Method Type: POST

Mandatory Parameters: accountid, devicename

Sample request

```
{
"request": {
"method":
"getApplicableMisconfigurationRemediation",
"parameters": {
"parameterset": [
{
"parameter": [
{
"key": "accountid",
"value": "Test Account"
},
{
"key": "device",
"values": [
"Test Device"
]
}
]}]}}}
```

Sample response

```
{
"remediationinfo": [
{
"account": "Test Account",
"assetpatchinfo": [
{
"assetname": "Microsoft Windows 11",
"displayname": "Microsoft Windows 11",
"remjobpatchinfo": [
{
"patchid": "cce-96571-5-patch.inf",
"eri": "eri:com.secpod.eri:21743",
"reboot": "FALSE",
"patch_size": "4.0 KiB",
"patch_releasedate": "2022-05-06",
"patch_version": "",
"fixinfo": "It is feasible for a attacker to
disguise a Trojan horse program as a printer
driver. The program may appear to users as
if they must use it to print, but such a
program could unleash malicious code on your
computer network. To reduce the possibility
of such an event, only administrators should
be allowed to install printer drivers.
However, because laptops are mobile devices,
laptop users may occasionally need to
install a printer driver from a remote
source to continue their work. Therefore,
this policy setting should be disabled for
laptop users, but always enabled for desktop
users.\n\n"
},
{
"patchid": "cce-96516-0-patch.inf",
"eri": "eri:com.secpod.eri:21729",
"reboot": "FALSE",
"patch_size": "4.0 KiB",
"patch_releasedate": "2022-05-06",
"patch_version": "",
"fixinfo": "This policy setting prevents
connected users from being enumerated on
domain-joined
computers.\n\n&lt;br&gt;&lt;br&gt;    If you
enable this policy setting, the Logon UI
will not enumerate any connected users on
domain-joined
computers.\n\n&lt;br&gt;&lt;br&gt;    If you
disable or do not configure this policy
setting, connected users will be enumerated
on domain-joined computers.\n\n"
}
]
}
]
```

Sample request	Sample response
	}] }

Possible Error Cases:

- Failed due to invalid account name.
- Service CM is not enabled.
- No devices present for current selection.
- No applicable missing patches /fixes.
- Invalid input.

Create Misconfiguration Remediation Job

This method is used to create remediation jobs for mis-configuration patches.

Method Name: createMisconfigurationRemediationJob

Method Type: POST

Mandatory Parameters: accountid, name, starttime, schedule, starttime, schedule, startwindowtime, device, and remjobpatchesinfo.

Sample Request	Sample Response
{ "request": { "method": "createMisconfigurationRemediationJob", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "TestAccount1" }, { "key": "name", "value": "MisConfRemJob1" }, { "key": "starttime", "value": "03:24:PM" }, { "key": "endtime", "value": "03:50:PM" }, { "key": "startwindowtime", "value": "03:30:PM" }, { "key": "forcereboot", "value": "true" }, { "key": "schedule",	{ "response": { "method": "createMisconfigurationRemediationJob", "results": { "result": [{ "key": "createRemediationJob", "status": "SUCCESS", "reason": "" }] } } }

<pre>"value": "2024-12-29" }, { "key": "device", "values": ["TestDevice1"] }, { "key": "remjobpatchesinfo", "valuesjson": [{ "assetname": "Ubuntu 18.04", "remjobpatchinfo": [{ "patchid": "cce-92627-9-patch.sh", "eri": "eri:com.secpod.eri:14173", "reboot": "FALSE" }] }] }] }] }}}</pre>	

Possible Error Cases:

- Failed due to invalid account name or id: <account>.
- Failed due to invalid task name.
- User/ Account need subscription to perform the operation.
- No devices present for current selection.
- Schedule value cannot be empty.
- Invalid schedule value.
- Start time cannot be empty.
- Start window time cannot be empty.
- Invalid starttime value.
- Invalid endtime value.
- Start and end time cannot be same.
- Invalid start window time value.
- Start window time should be lesser than endtime.
- Invalid forcereboot value.
- Invalid Input:autoreboottime.
- No applicable devices found for selected patches. Task <task name> creation failed!
- Invalid Input.

Get Non-Security Job Details

This method fetches the details of a non-security remediation , firmware, rollback and reboot jobs in Saner CM. Details such as ‘schedule’, ‘start time’, ‘end time’, ‘startwindowtime’, ‘remtype’, ‘startmessage’, and ‘endmessage’ are fetched.

You need to provide a unique job name while creating an IR job in Saner CM. You can’t give the job name as PM or CM.

Method Name: getNonSecJobDetails

Method Type: POST

Mandatory Parameters: name and accountid

Sample request	Sample response
<pre>{ "request": {</pre>	<pre>For Rollback { "IRDetails": [</pre>

Sample request	Sample response
<pre>{ "method": "getnonsecjobdetails", "parameters": { "parameterset": [{ "parameter": [{ "key": "name", "value": "non_sec_job" }, { "key": "accountid", "value": "Test_Account" }] }] } } }</pre>	<pre>{ "rollback": [{ "name": "cce-95046-9-patch.sh", "assetname": "cce-95046-9-patch.sh", "type": "original" }], "schedule": "immediate", "autoreboot": "false", "forcereboot": "false", "account": "Test_Account" }] }</pre>
	<pre>{ "IRDetails": [{ "action": "systemreboot", "values": [{ "key": "message", "value": "Your system will be re-booted shortly" }, { "key": "timeinminutes", "value": "0" }], "type": "system", "account": "NewAccount" }] }</pre>

Possible Error Cases

- Invalid Input.
- Failed due to invalid account name or id.
- User/Account need subscription to perform operation.
- Failed due to invalid task name.

Get Applicable Misconfiguration Rollback Patches

This method is used to get a list of all installed misconfiguration patches applicable for the rollback

Method Name: getMisconfigurationFixForRollback
Method Type: POST

Mandatory Parameters: accountid,

Sample Request	Sample Response
<pre>{ "request": { "method": "getMisconfigurationFixforRollback", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "API_testing" }</pre>	<pre>{ "remediationinfo": [{ "account": "ACC_22Dec", "devicerollbackpatchinfo": [{ "devicename": "desktop- d04sj8i", "rollbackpatchinfo": [{ "patchid": "KB5021089",</pre>

```
      ]
     }
    ]
   }
  }
}
```

```
       "asset": "2022-12
Cumulative Update for .NET
Framework 3.5, 4.8 and 4.8.1
for Windows 10 Version 22H2 for
x64 (KB5021089)",
       "size": "67.7 MiB",
       "rollbackstatus":
"TRUE",
       "installeddate": "2022-
12-23 07:01:57 AM UTC"
      },
      {
       "patchid": "KB5012170",
       "asset": "2022-12
Security Update for Windows 10
Version 22H2 for x64-based Sys-
tems (KB5012170)",
       "size": "116.2 KiB",
       "rollbackstatus":
"TRUE",
       "installeddate": "2022-
12-23 07:01:57 AM UTC"
      }
     ]
    },
    {
     "devicename": "win-
o5bmvc1lpno",
     "rollbackpatchinfo": [
      {
       "patchid": "KB2898871",
       "asset": "Security
Update for Mi-crosoft .NET
Framework 4.5.1 on Windows 8.1
and Windows Server 2012 R2 for
x64-based Systems (KB2898871)",
       "size": "35.3 MiB",
       "rollbackstatus":
"TRUE",
       "installeddate": "2022-
12-23 08:21:47 AM UTC"
      }
     ]
    }
   ]
  }
 ]
}
```


Possible Error Cases

- Failed due to invalid account name or Id : <account>
- Invalid Input.
- Service CM is not enabled.
- No applicable devices found.
- No Records.

Create Misconfiguration Rollback Patches

This method allows you to roll back the installed misconfiguration patches for a group. 'Account ID' is the required field.

Method Name: createMisconfigurationRollbackTask
Method Type: POST

Mandatory Parameters: accountid, name, starttime, startwindowtime, device, schedule, and rollbackpatchesinfo.

Sample Request

```
{
 "request": {
  "method": "createMisconfigurationRoll-
backTask",
  "parameters": {
   "parameterset": [
    {
     "parameter": [
      {
       "key": "accountid",
       "value": "API_testing"
      },
      {
       "key": "name",
       "value": "MyRollbackJob1"
      },
      {
       "key": "schedule",
       "value": "2023-09-24"
      },
      {
       "key": "starttime",
       "value": "15:00:00"
      },
      {
       "key": "startmessage",
       "value": "Remediation starting"
      },
      {
       "key": "endmessage",
       "value": "Remediation Finished"
      },
      {
       "key": "startwindowtime",
       "value": "15:24:PM"
      },
      {
       "key": "autoreboot",
       "value": "TRUE"
      },
      {
       "key": "forcereboot",
       "value": "FALSE"
      },
      {
       "key": "rebootmessage",
       "value": "Rebooting in 2minutes"
      },
      {
       "key": "autoreboottime",
       "value": "2023-09-25.18:00:00"
      },
      {
       "key": "endtime",
       "value": "18:30:00"
      },
      {
       "key": "rollbackpatchesinfo",
       "valuesjson": [
        {
         "devicename": "desktop-1",
         "rollbackpatchinfo": [
          {
           "patchid": "cce-91967-0-
patch.sh"
```

Sample Response

```
{
 "response": {
  "method":
"createMisconfigurationRollbackTask",
  "results": {
   "result": [
    {
     "key":
"createMisconfigurationRollbackTask",
     "status": "SUCCESS",
     "reason": ""
    }
   ]
  }
 }
}
```

<table>
<tr>
<td><pre>
            },
            {
             "patchid": "cce-91977-9-patch.sh
"
            }
           ]
          }
         ]
        }
       ]
      }
     ]
    }
   }
}
</pre></td>
<td></td>
</tr>
</table>

Possible Error Cases

- Schedule value cannot be empty.
- Invalid schedule value.
- Start time cannot be empty.
- Start window time cannot be empty.
- Invalid forcereboot value.
- Invalid start window time value.
- Invalid starttime value.
- Invalid endtime value.
- Start window time should be lesser than end time.
- Start and end time cannot be same.

Get Misconfiguration Rollback Status

This method gets the status of the job/task and related info related to the misconfiguration rollback status job.

Method Name: getMisconfigurationRollbackStatus
Method Type: POST

Mandatory Parameters: accountid and name

<table>
<tr><th>Sample request</th><th>Sample response</th></tr>
<tr>
<td><pre>
{
 "request": {
  "method":
"getmisconfigurationrollbackstatus",
  "parameters": {
   "parameterset": [
    {
     "parameter": [
      {
       "key": "name",
       "value": "rolllback"
      },
      {
       "key": "accountid",
       "value": "{{AccountName}}"
      },
      {
       "key": "device",
       "values": [
        "{{hostname}}"
</pre></td>
<td><pre>
{
 "RollbackStatus": [
  {
   "accountrollbackstatus": [
    {
     "JobName": "rollback",
     "CreationTime": "2023-08-04 06:55:24 AM
UTC",
     "Groupname": "oracle linux",
     "Hostname": "qa-oracle-linux-8.3-x64",
     "LastUpdate": "2023-08-04 06:55:47 AM
UTC",
     "ActualDate": "Saner Agent: 2023-08-04
06:55:47 AM UTC",
     "jobType": "IR",
     "PatchName": "cce-94413-2-patch.sh",
     "Status": "success",
     "Reason": ""
    },
    {
</pre></td>
</tr>
</table>

Sample request	Sample response
<pre>] }, { "key": "group", "values": ["oracle linux"] }, { "key": "family", "values": ["unix"] }] }] } } }</pre>	<pre>"JobName": "rollback", "CreationTime": "2023-08-04 06:55:24 AM UTC", "Groupname": "oracle linux", "Hostname": "qa-oracle-linux-8.3-x64", "LastUpdate": "2023-08-04 06:55:47 AM UTC", "ActualDate": "Saner Agent: 2023-08-04 06:55:47 AM UTC", "jobType": "IR", "PatchName": "cce-94494-2-patch.sh", "Status": "success", "Reason": "" }, { "JobName": "rollback", "CreationTime": "2023-08-04 06:55:24 AM UTC", "Groupname": "oracle linux", "Hostname": "qa-oracle-linux-8.3-x64", "LastUpdate": "2023-08-04 06:55:47 AM UTC", "ActualDate": "Saner Agent: 2023-08-04 06:55:47 AM UTC", "jobType": "IR", "PatchName": "cce-94511-3-patch.sh", "Status": "success", "Reason": "" }], "account": "AlltoolsEnabled" }] }</pre>

Possible Error Case

- Invalid input.
- Failed due to invalid account name or Id: <account>.
- Invalid Input: <job name>.

Delete Misconfiguration Rollback Task

This method is used to delete any existing misconfiguration rollback task.

Method Name: deleteMisconfigurationRollbackTask
Method Type: POST

Mandatory Parameters: accountid and name

Sample request	Sample response
<pre>{ "request": { "method": "deletemisconfigurationrollbacktask", "parameters": { "parameterset": [{ "parameter": [{ "key": "name", "value": "rollback"</pre>	<pre>{ "response": { "method": "deletePatchRollbackTask", "results": { "result": [{ "key": "rollback", "status": "SUCCESS", "reason": "" }]</pre>

Sample request	Sample response
<pre>}, { "key": "accountid", "value": "{{AccountName}}" }] }] } } }</pre>	<pre>} } }</pre>

Possible Error Cases

- Invalid Input.
- Failed due to invalid task name.
- Failed due to invalid account name or id.

Get Automation Rule Status

This method returns whether the automation rule in Saner CM (Compliance Management) is paused or running. If the rule name is not specified, the API will return the results for all the rules.

Method Name: getAutomationRuleStatus

Method Type: POST

Mandatory Parameters: accountname

Sample request	Sample response
<pre>{ "request": { "method": "getAutomationRuleStatus", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountname", "value": "Test_Account" }, { "key": "rulenames", "values": ["automationTest","test rule"] }] }] } } }</pre>	<pre>In case of success: { "automationRuleStatusInfo": [{ "rule_name": "automationTest", "rule_status": "running" }, { "rule_name": "test rule", "rule_status": "paused" }] }</pre>

Possible Error Cases

- Missing mandatory fields. Account name must be provided
- Field key cannot be empty
- Invalid rule name
- Failed due to invalid account name or Id

- Field key is invalid
- Account name must be provided
- Remediation rules {rule names} do not exist
- Failed to get automation rule status.

Update Automation Rule Status

This method allows you to pause or resume a remediation rule in Saner CM (Compliance Management). The key “status” accepts ‘resume’ and ‘pause’ as values.

.

Method Name: updateAutomationRuleStatus

Method Type: POST

Mandatory Parameters: accountname, rulenames, and status

Sample request	Sample response
<pre>{ "request": { "method": "updateAutomationRuleStatus", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountname", "value": "Test_Account" }, { "key": "rulenames", "values": ["automationTest", "automationTest2"] }, { "key": "status", "value": "pause" }] }] } } }</pre>	<pre>In case of success: { "response": { "method": "updateAutomationRuleStatus", "results": { "result": [{ "key": "updateAutomationRUleStatus", "status": "success", "reason": "Updated remediation automation rules successfully, one or more rules were already in paused state" }] } } }</pre>
<pre>---------- To resume an automation rule: { "request": { "method": "updateAutomationRuleStatus", "parameters": { "parameterset": [</pre>	<pre>---------- In case of success: { "response": { "method": "updateAutomationRuleStatus", "results": { "result": [{</pre>

Sample request	Sample response
<pre>{ "parameter": [{ "key": "accountname", "value": "Test_Account" }, { "key": "rulenames", "values": ["automationTest"] }, { "key": "status", "value": "resume" }] }] } } }	<pre>"key": "updateAutomationRuleStatus", "status": "success", "reason": "Updated remediation automation rule successfully" }] } } }

Possible Error Cases

- Missing mandatory fields. Account name, rule name and status must be provided
- Field key cannot be empty
- Invalid rule name {name}
- Invalid status
- Invalid value. Field status can be either pause or resume
- Failed due to invalid account name or Id
- Field key is invalid
- Account name must be provided
- Status must be provided
- Rule names must be provided
- Remediation rules {rule names} do not exist
- Failed to update automation rule

Risk Prioritization

Saner Risk Prioritization helps you identify and prioritize the most severe vulnerabilities and misconfigurations in your Organization's IT infrastructure that need immediate attention. This section provides insights into the Saner Risk Prioritization APIs and their usage.

Get Mission-Critical Device Data

This method provides the details of assets and devices configured as mission-critical.

Method Name: getMissionCriticalDeviceData

Method Type: POST
Mandatory Parameters: organization and account_name.

URL: https://saner.secpod.com/RPWebService/getMissionCriticalDeviceData

Sample request	Sample response
<pre>{ "organization": "<Org name>", "account_name": "<account name>" }</pre>	<pre>{ "critical_software_assets": ["Adobe InDesign x64", "JetBrains IntelliJ IDEA x86"], "decision_questions": {}, "device_tags": { "business_centric": ["sp-test-laptop"], "data_centric": ["sp-test1-desktop"], "internet_facing": ["sp-test2-lap"] }, "high_devices": ["sp-test4-desktop", "sp-test5-desktop", "sp-test6-laptop"], "low_devices": [], "medium_devices": [] }</pre>

Possible Error Cases

- Invalid Organization Name
- Invalid Account Name
- Invalid User id
- RP not enabled.

Save Mission Critical Device Data

This method is used to save critical assets, critical device, and configured questions.

Method Name: saveMissionCriticalDeviceData

Method Type: POST

Mandatory Parameters: organization and account_name.

URL: https://saner.secpod.com/RPWebService/saveMissionCriticalDeviceData

Sample request	Sample response
<pre>{ "organization": "<Org name>", "account_name": "<account name>", "details": {</pre>	<pre>{ "status": "Success" }</pre>

Sample request	Sample response
"low_devices": [], "high_devices": ["sp-test-desktop", "sp-test1-desktop", "sp-test2-laptop"], "medium_devices": [], "device_tags": { "business_centric": ["sp-test3-laptop"], "data_centric": ["sp-test4-desktop"], "internet_facing": ["sp-test5-lap"] }, "decision_questions":{ "M1017": "yes"}, "critical_software_assets": ["Adobe InDesign x64", "JetBrains IntelliJ IDEA x86"] } }	

Possible Error Cases

- Invalid Organization Name
- Invalid Account Name
- Invalid User id.
- RP not enabled.

Get Risk Prioritization Summary

This method provides the count of risks under the Act, Attend, Track*, and Track categories.

Track stands for No action required, remediation tracking within standard update timelines.
Track* signifies that vulnerability has specific characteristics, requires closer monitoring for changes, remediation tracking within standard update timelines.
Attend stands for vulnerabilities that require attention from organization's internal; supervisory level individuals, necessary actions may include required assistance or information about vulnerability, involve a notification internally or externally (Attend Sooner).
Act categorizes vulnerabilities that requires attention. Act as soon as possible.

Method Name: getRiskPrioritizationSummary

Method Type: POST

Mandatory Parameters: organization and account_name

URL: https://saner.secpod.com/RPWebService/getRiskPrioritizationSummary

Sample request	Sample response
<pre>{ "organization": "<Org name>", "account_name": "<account name>" }</pre>	<pre>{ "act": 119, "attend": 92, "track": 2265, "track_star": 145 }</pre>

Possible Error Cases

- Invalid Organization Name
- Invalid Account Name
- Invalid User id.
- RP not enabled.

Get Risk Summary

This method gives a summary of the risk associated with risk id.

Method Name: getRiskSummary

Method Type: POST

Mandatory Parameters: organization, account_name, and risk id

URL: https://saner.secpod.com/RPWebService/getRiskSummary

Sample request	Sample response
<pre>{ "organization": "Test Org", "account_name": "Test Account", "risk_id": "CVE-2022-32917", "priority": "Act", "asset_name": "Apple Mac OS 11", "automatable": "yes", "mission_critical": "low" }</pre>	<pre>{ "rpresponse": [{ "affected_devices": ["sp-test1-laptop", "sp-test2-laptop", "sp-test19-lap"], "asset_name": "Google Chrome x64", "authentication_disclosure": "unknown", "automatable": "no", "availability": "HIGH", "category": "heap corruption", "chainable_devices": [], "chainable_risks": { "sp-test2-laptop": "CVE-2022-3656 and 7 more", "sp-test9-laptop": "CVE-2022-3656 and 7 more" }, "chaining": true, "confidentiality": "HIGH", "critical_vendors": ["google"], "crucial_findings": ["Vendor Advisory", "Google Chromium V8 Type Confusion Vulnerability(CISA)", "Google Project Zero"</pre>

Sample request	Sample response


```
        ],
        "device_having_effective_barriers": [],
        "device_tags": {},
        "enumerable_on_network": false,
        "exploit_prevention": [],
        "exploit_prevention_enabled": false,
        "exploitability": "high",
        "exploitable": "high",
        "exploitation_score": 8.32,
        "exploitation_status": true,
        "fix_info": "Security update for Google
Chrome on Windows.Security update for Google Chrome
on Linux",
        "high_devices": [
            "sp-test8-laptop",
             "sp-test6-laptop"
             ],
        "integrity": "HIGH",
        "internet_facing_devices": [],
        "low_devices": [],
        "medium_devices": [],
        "mission_critical": "high",
        "modified_date": "2023-05-03
12:16:00+00:00",
        "other_references": [],
        "priority": "Act",
        "published_date": "2022-11-01
23:15:00+00:00",
        "risk_id": "CVE-2022-3723",
        "risk_references": [
            "https://crbug.com/1378239",

"https://chromereleases.googleblog.com/2022/10/stab
le-channel-update-for-desktop_27.html",

"https://security.gentoo.org/glsa/202305-10"
        ],
        "risk_score": 8.8,
        "risk_type": "VULNERABILITY",
        "technical_impact": "total",
        "technicality": "heap corruption",
        "title": "A Vulnerability is discovered in
Type confusion and V8 in Google Chrome",
        "user_interaction_requirement": true,
        "v2": {
            "Access Complexity": "",
            "Access Vector": "",
            "Authentication": "",
            "Availability": "",
            "CVSS Score": 0,
            "Confidentiality": "",
            "Exploit Score": 0,
            "Impact Score": 0,
            "Integrity": "",
            "Vector": ""
        },
        "v3": {
            "Attack Complexity": "LOW",
            "Attack Vector": "NETWORK",
            "Availability": "HIGH",
            "CVSS Score": 8.8,
            "Confidentiality": "HIGH",
            "Exploit Score": 2.8,
            "Impact Score": 5.9,
            "Integrity": "HIGH",
            "Privileges Required": "NONE",
            "Scope": "UNCHANGED",
```

Sample request	Sample response
	<pre>"User Interaction": "REQUIRED", "Vector": "CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H" }, "version": "3" }] }</pre>

Possible Error Cases

- Invalid Organization Name
- Invalid Account Name
- Invalid User id.
- RP not enabled.
- Bad request if risk id is not a string.
- Invalid input if riskid is not given.

Get Risk Details

This method lists the risks based on Priority (Act/ Attend/ Track*/ Track), also includes risk information such as ID, Title, Affected Products, Affected Devices, Mission Prevalence, Exploitation Status, Automatable, Technical Impact, and Fix details. In this API, specifying '*row_limit:0'* will fetch all the entries.

Method Name: getRiskDetails

Method Type: POST

Mandatory Parameters: organization and account_name

URL: https://saner.secpod.com/RPWebService/getRiskDetails

Sample request	Sample response
<pre>{ "organization": "Test_Org", "account_name": "Test_Account", "row_limit": 0 }</pre>	<pre>{ "rpresponse": [{ "affected_devices": ["sp-test-laptop"], "asset_name": "Apple Mac OS 11", "automatable": "no", "exploitable": "high", "fix_info": "This is an OS remediation.\nSaner will install all latest updates for this product\nsilently. It is recommended that you do not power off machine/disconnect\nnetwork when this is going on.", "mission_critical": "high", "priority": "Act", "risk_id": "CVE-2022-32894", "technical_impact": "total", "title": "An out-of-bounds write issue was addressed with improved bounds checking. This issue is fixed in iOS 15.6.1 and iPadOS 15.6.1, macOS Monterey 12.5.1. An application may be able to execute arbitrary code with kernel privileges. Apple is aware of a report that this issue may have been actively exploited." }, ...]</pre>

Sample request	Sample response
	}

Possible Error Cases

- Invalid Organization Name
- Invalid Account Name
- Invalid User id.
- RP not enabled.
- Bad request if ascending is not Boolean value or row limit, current page is not numerical values or sort, search are not strings or filter is not dictionary or required fields are not listed.

Get Risk Mitigation Details

This method gives tactics and techniques, count of act, attend, track, and track*, asset names, and affected devices for each mitigation id. Here, in this API, specifying '*row_limit:0'* will fetch all the entries.

Method Name: getRiskMitigationDetails

Method Type: POST

Mandatory Parameters: organization, and account_name

URL: https://saner.secpod.com/RPWebService/getRiskMitigationDetails

Sample request	Sample response
{ "organization": "Test_Org", "account_name": "Test_Account", "row_limit": 0 }	{ "rpresponse": [{ "Act": 0, "Attend": 0, "Track": 1, "Track*": 0, "affected_devices": ["qa-centos-7core-x64"], "affected_devices_count": 1, "asset_count": 1, "asset_names": ["kernel"], "fix_info": [], "fix_info_count": 0, "mitigation": "M1037", "non_affected_devices": [], "non_affected_devices_count": 0, "not_evaluated_devices": [], "not_evaluated_devices_count": 0, "patches": [], "risk_id": "CVE-2020-11668", "risk_type": "VULNERABILITY", "tactics": ["TA0040"], "tactics_count": 1, "techniques": ["T1499.004"

Sample request	Sample response
	<pre>], "techniques_count": 1, "title": "A Vulnerability is discovered in Linux kernel before 5.6.1" }] }</pre>

Possible Error Cases

- Invalid Organization Name
- Invalid Account Name
- Invalid User id.
- RP not enabled.
- Bad request if ascending is not Boolean value or row limit, current page is not numerical values or sort, search are not strings or filter is not dictionary or required fields are not list

Get Critical Asset Risks Details

This method is used to get a tabular listing of critical software assets based on Priority (Act/ Attend/ Track*/ Track) and affected device count. Here, in this API, specifying '*row_limit:0'* will fetch all the entries.

Method Name: **getCriticalAssetRiskDetails**
Method Type: POST

Mandatory Parameters: organization, and account_name

URL: https://saner.secpod.com/RPWebService/getCriticalAssetRiskDetails

Sample request	Sample response
<pre>{ "organization": "Test_Org", "account_name": "Test_Account", "row_limit": 0 }</pre>	<pre>{ "rpresponse": [{ "Act": 5, "Attend": 0, "Track": 75, "Track*": 207, "affected_devices": ["sp-test1-laptop", "sp-test2-laptop", "sp-test3-laptop", "sp-test4-desktop", "sp-test5-laptop"], "asset_name": "Microsoft Windows 10 21h2 x64", "device_count": 5 }....] }</pre>

Possible Error Cases

- Invalid Organization Name
- Invalid Account Name
- Invalid User id.
- RP not enabled.

- Bad request if ascending is not Boolean value or row limit, current page is not numerical values or sort, search are not strings or filter is not dictionary.

Get Risk Automatability Details

This method is used to get the details about a vulnerability – whether it is automatable or not? A significant attribute of the Risk Prioritization algorithm is Automatable (no/yes). Automatable represents the ease and speed with which a cyber threat actor can cause exploitation events. Automatable captures the answer to the question, “Can an attacker reliably automate, creating exploitation events for this vulnerability?”. Here, in this API, specifying ‘*row_limit:0’* will fetch all the entries.

Method Name: **getRiskAutomatabilityDetails**
Method Type: POST

Mandatory Parameters: organization, and account_name

URL: https://saner.secpod.com/RPWebService/getRiskAutomatabilityDetails

Sample request	Sample response
{ "organization": "Test_Org", "account_name": "Test_Account", "row_limit": 0 }	{ "rpresponse": [{ "automatable": "no", "chaining": true, "enumerable_on_network": false, "exploit_prevention": [], "exploit_prevention_enabled": false, "priority": "Track*", "risk_id": "CVE-2021-29967", "risk_score": 8.8, "title": "Memory corruption and arbitrary code. in Mozilla developers reported memory safety bugs present and Firefox" }] }

Possible Error Cases

- Invalid Organization Name
- Invalid Account Name
- Invalid User id.
- RP not enabled.
- Bad request if ascending is not Boolean or row limit, the current page is not numerical values or sort, search are not strings or filter is not dictionary or required fields are not listed.

Get Risk Technical Impact Details

This method is used to get the technical impact of a vulnerability. Technical impact is similar to the Common Vulnerability Scoring System (CVSS) base score’s concept of “severity.” When evaluating technical impact, the definition of scope is particularly important. The decision point, "Total," is relative to the affected component where the vulnerability resides. If a vulnerability discloses authentication or authorization credentials to the system, this information disclosure should also be scored as “Total” if those credentials give an adversary total control of the component.

Method Name: **getRiskTechnicalImpactDetails**

Method Type: POST

Mandatory Parameters: organization, and account_name

URL: https://saner.secpod.com/RPWebService/getRiskTechnicalImpactDetails

Sample request	Sample response
<pre>{ "organization": "Test_Org", "account_name": "Test_Account", "row_limit": 0 }</pre>	<pre>{ "rpresponse": [{ "authentication_disclosure": "unknown", "availability": "HIGH", "confidentiality": "HIGH", "integrity": "HIGH", "priority": ["Track*"], "risk_id": "CCE-41475-5", "risk_score": 6.8, "technical_impact": "total", "technicality": "unknown", "title": "Disable: 'Recovery console: Allow automatic administrative logon' for securitylevel\n\nThe recovery console is a command-line environment that is used to recover from system problems. If you enable this policy setting, the administrator account is automatically logged on to the recovery console when it is invoked during startup.\n\n Counter Measure:\n Disable the Recovery Console: Allow automatic administrative logon setting.\n\n Potential Impact:\n Users will have to enter a user name and password to access the Recovery Console." }, ...] }</pre>

Possible Error Cases

- Invalid Organization Name
- Invalid Account Name
- Invalid User id.
- RP not enabled.
- Bad request if ascending is not Boolean or row limit, current page is not numerical values or sort, search are not strings or filter is not dictionary or required fields are not listed.

Get Risk Exploitability Details

This method is used to get the exploitability details of a vulnerability. Exploitation determines the present state of exploitation of the vulnerability. It does not predict future exploitation or measure feasibility or ease of adversary development of future exploit code; rather, it acknowledges available information at time of analysis. As the current state of exploitation often changes over time, answers are timestamped. Sources that can provide public reporting of active exploitation include the vendor's vulnerability notification, SecPod's Malware Vulnerability Enumeration, Google Project Zero, CISA KEVs,
the National Vulnerability Database (NVD) and links therein and reliable threat reports that list either the CVE-ID or common name of the vulnerability.

Method Name: **getRiskExploitabilityDetails**
Method Type: POST

Mandatory Parameters: organization, and account_name

URL: https://saner.secpod.com/RPWebService/getRiskExploitabilityDetails

Sample request	Sample response
{ "organization": "Test_Org", "account_name": "Test_Account", "row_limit": 0 }	{ "rpresponse": [{ "asset_name": ["OpenSSH"], "category": "Exposure of Sensitive Information to an Unauthorized Actor", "critical_vendors": ["openbsd"], "crucial_findings": ["Third Party Advisory"], "exploitability": "low", "exploitation_score": 0.0, "modified_date": "2021-04-01 15:32:00+00:00", "priority": ["Track"], "published_date": "2007-05-21 20:30:00+00:00", "risk_id": "CVE-2007-2768", "risk_score": 4.3, "title": "A vulnerability in the PAM (Physical Account Management) component of OpenSSH can be exploited by attackers to gain access to a victim's account." }, ...] }

Possible Error Cases

- Invalid Organization Name
- Invalid Account Name
- Invalid User id.
- RP not enabled.
- Bad request if ascending is not Boolean or row limit, current page is not numerical values or sort, search are not strings or filter is not dictionary or required fields are not listed.

Get Device Risk Details

This method returns a tabular listing of critical devices based on Priority (Act/Attend/Track*/Track), and it also includes device details such as Primary OP address and group. Here, in this API, specifying '*row_limit:0'* will fetch all the entries.

Method Name: **getDeviceRiskDetails**
Method Type: POST

Mandatory Parameters: organization,and account_name.

URL: https://saner.secpod.com/RPWebService/getDeviceRiskDetails

Sample request	Sample response
{ "organization": "Test_Org", "account_name": "Test_Account", "row_limit": 0 }	{ "rpresponse": [{ "Act": 4, "Attend": 1, "DeviceStatusDisplaytext": "Not seen since 238 days, Last Seen: 2022-10-31 10:25:57 AM UTC",

Sample request	Sample response
	"Track": 284, "Track*": 168, "business_centric": false, "critical_device": "essential", "data_centric": false, "device-type": "end-point", "family": "windows", "group": "windows 10", "high": true, "hostname": "sp-test-laptop", "internet_facing": false, "ip-address": "192.168.x.x", "low": false, "mac-address": "F0-XX-XX-XX-XX-XX", "medium": false, "os": "Microsoft Windows 10 v21H2 architecture 64-bit" }] }

Possible Error Cases

- Invalid Organization Name
- Invalid Account Name
- Invalid User id.
- RP not enabled.
- Bad request if ascending is not Boolean or row limit, current page is not numerical values or sort, search are not strings or filter is not dictionary or required fields are not listed.

Get RP Json

This method is used to fetch all the information about risk in a CycloneDX format.

Method Name: **getRPJson**
Method Type: POST

Mandatory Parameters: organization, account_name, and risk_id

URL: https://saner.secpod.com/RPWebService/getRPJson

Sample request	Sample response
{ "organization": "<Org name>", "account_name": "<account name>", "risk_ids": [] }	{ "bomFormat": "CycloneDX", "specVersion": "1.4", "version": 1, "vulnerabilities": [{ "advisories": [{ "title": "Patch through Saner", "url": "https://192.168.3.221/modules/PM/PMcontrol.jsp?command=patchmanagement" }], "affects": [{ "ref": ["https://crbug.com/1378239", "https://chromereleases.googleblog.com/2022/10/stable-channel-update-for-desktop_27.html", "https://security.gentoo.org/glsa/202305-10"]

Sample request	Sample response
	}], "analysis": [{ "asset name": "Google Chrome x64", "automatable": "no", "details": "heap corruption", "mission critical": "high", "priority": "Act", "technical impact": "total" }, { "asset name": "Google Chrome x86", "automatable": "no", "details": "heap corruption", "mission critical": "high", "priority": "Act", "technical impact": "total" }, { "asset name": "Google Chrome Enterprise x64", "automatable": "no", "details": "heap corruption", "mission critical": "high", "priority": "Act", "technical impact": "total" }], "description": "A Vulnerability is discovered in Type confusion and V8 in Google Chrome", "id": "CVE-2022-3723", "published": "2022-11-01 23:15:00+00:00", "ratings": [{ "method": "CVSSV3", "score": 8.8, "severity": "High", "source": { "name": "NVD", "url": "https://nvd.nist.gov/vuln/detail/CVE-2022-3723" }, "vector": "CVSS:3.1/AV:N/AC:L/PR:N/UI:R/S:U/C:H/I:H/A:H" }], "recommendation": "Security update for Google Chrome on Windows.Security update for Google Chrome on Linux", "source": { "name": "SCAPrepo", "url": "https://www.scaprepo.com/control.jsp?command=search&search=CVE-2022-3723" }, "updated": "2023-05-03 12:16:00+00:00" }] }

Possible Error Cases

- Invalid Organization Name
- Invalid Account Name
- Invalid User id.
- RP is not enabled.
- Bad request if Risk ids are not a list of strings.

Get Risk Prioritization Status

This method is used to get the last scan time and status of the RP Scan.

Method Name: **getRiskPrioritizationStatus**
Method Type: POST

Mandatory Parameters: organization, and account_name.

URL: https://saner.secpod.com/RPWebService/getRiskPrioritizationStatus

Sample request	Sample response
{ "organization": "Test Org", "account_name": "Test Account" }	{ "lastScan": "2023-04-11 19:32:57.787923", "status": "RP Scan is Ongoing" } OR { "lastScan": "2023-04-12 08:03:51.055", "status": "Completed" } OR { "lastScan": "", "status": "Schedule Risk Prioritization" }

Possible Error Cases

- Invalid Organization Name
- Invalid Account Name
- Invalid User id.
- RP not enabled.

Get Risk On Mission Critical

This method shows the percentage of essential devices for business mission prevalence that are at risk.

Method Name: **getRiskonMissionCritical**
Method Type: POST

Mandatory Parameters: organization, and account_name.

URL: https://saner.secpod.com/RPWebService/getRiskonMissionCritical

Sample request	Sample response
{ "organization": "Test Org", "account_name": "Test Account" }	{ "rpresponse": [{ "affected_devices": 10, "unaffected_devices": 5 }] }

Possible Error Cases

- Invalid Organization Name
- Invalid Account Name
- Invalid User id.
- RP not enabled.

Get Risk On Mission Prevalence

This method gives a summary of risk count on devices based on Mission Prevalence.

Method Name: **getRiskonMissionPrevalence**
Method Type: POST

Mandatory Parameters: organization, and account_name.

URL: https://saner.secpod.com/RPWebService/getRiskonMissionPrevalence

Sample request	Sample response
<pre>{ "organization": "Test Org", "account_name": "Test Account" }</pre>	<pre>{ "rpresponse": [{ "device_type": "Essential Devices", "essential": "2155 Risks on 3 Essential Devices" }, { "device_type": "Support Devices", "support": "No Support Devices Configured" }, { "device_type": "Minimal Devices", "minimal": "192459 Risks on 508 Minimal Devices" }] }</pre>

Possible Error Cases

- Invalid Organization Name
- Invalid Account Name
- Invalid User id.
- RP not enabled.

Get Risk On Essential Devices

This method provides the summary of risk count essential devices that are categorized as Business Centric, Data Storage and Public Facing. Identifying such MEFs is part of business continuity planning and crisis planning.

Method Name: **getRiskonEssentialDevices**
Method Type: POST

Mandatory Parameters: organization, and account_name.

URL: https://saner.secpod.com/RPWebService/getRiskonEssentialDevices

Sample request	Sample response
{	{

Sample request	Sample response
<pre>{ "organization": "Test Org", "account_name": "Test Account" }</pre>	<pre>"rpresponse": [{ "business_centric": "0 Risks on 0 Business Centric Devices", "device_type": "Business Centric Devices" }, { "data_centric": "0 Risks on 0 Data Centric Devices", "device_type": "Data Centric Devices" }, { "device_type": "Internet Facing Devices", "internet_facing": "8 Risks on 1 Internet Facing Devices" }] }</pre>

Possible Error Cases

- Invalid Organization Name
- Invalid Account Name
- Invalid User id.
- RP not enabled.

Risk Prioritization

This method is used to perform an RP Scan on an Account.

Method Name: **prioritizeRisks**
Method Type: POST

Mandatory Parameters: organization, and account_name.

URL: https://saner.secpod.com/RPWebService/prioritizeRisks

Sample request	Sample response
<pre>{ "organization": "Test Org", "account_name": "Test Account" }</pre>	<pre>{ "status": "success" }</pre>

Possible Error Cases

- Invalid Organization Name
- Invalid Account Name
- Invalid User id.
- RP not enabled.

Get RP Trends

This method is used to determine how many risks and affected devices fall into act, attend, track, and track* for the last 30 scans. Multiple scans performed during the day will be considered a single scan.

Method Name: **getRPTrends**
Method Type: POST

Mandatory Parameters: organization, and account_name.

URL: https://saner.secpod.com/RPWebService/getRPTrends

Sample request	Sample response
<pre>{ "organization": "Test Org", "account_name": "Test Account" }</pre>	<pre>{ "rpresponse": [{ "act_devices": [], "act_devices_count": 1, "act_risk_count": 1, "act_risk_ids": [], "attend_devices": [], "attend_devices_count": 1, "attend_risk_count": 1, "attend_risk_ids": [], "track*_devices": [], "track*_devices_count": 1, "track*_risk_count": 1, "track*_risk_ids": [], "track_devices": [], "track_devices_count": 1, "track_risk_count": 1, "track_risk_ids": [] }] }</pre>

Possible Error Cases

- Invalid Organization Name
- Invalid Account Name
- Invalid User id.
- RP not enabled.

Get Chainable Risks

This method is used to get each device's list of chainable risks.

Method Name: **getChainableRisks**
Method Type: POST

Mandatory Parameters: organization, account_name, risk_id, and asset_name

URL: https://saner.secpod.com/RPWebService/getChainableRisks

Sample request	Sample response
<pre>{ "organization": "<Org name>", "account_name": "<account name>", "risk_id": "<risk id>", "asset_name": "<asset name>", "priority": "<priority Act/Attend/Track*/Track>", "automatable": "<automatability yes/no>",</pre>	<pre>{ "chainable_risk_detail": [{ "chainable_risks": ["CVE-2021-44228"], "device_name": "sp-test-laptop" }, { "chainable_risks": ["CVE-2021-44228"],</pre>

Sample request	Sample response
<pre>"mission_critical": "<mission critical high/low/medium>" }</pre>	<pre>"device_name": "sp-test1-laptop" }, { "chainable_risks": ["CVE-2021-44228"], "device_name": "sp-test2-lap" }, { "chainable_risks": ["CVE-2021-44228"], "device_name": "sp-test3-lap" }] }</pre>

Possible Error Cases

- Invalid Organization Name
- Invalid Account Name
- Invalid User id.
- RP not enabled.

Get MVE Details

This method is used to get the details of the MVE.

Method Name: **getMVEDetails**
Method Type: POST

Mandatory Parameters: organization, account_name, and mve_id

URL: https://saner.secpod.com/RPWebService/getMVEDetails

Sample request	Sample response
<pre>{ "organization": "Test Org", "account_name": "Test Account", "mve_id": "MVE-000360" }</pre>	<pre>{ "rpresponse": [{ "threat_info": [{ "cve_ids": [{ "affected_os": ["Windows"], "affected_products": ["Microsoft's Authenticode Digital Signing Package in Microsoft Windows"], "cve_id": ["CVE-2020-1464"], "impact": ["Arbitrary Code Execution"], "references": { "advisory": ["https://portal.msrc.microsoft.com/en-US/security-guidance/advisory/CVE-2020-1464"], "cve_reference": [</pre>

Sample request	Sample response


```
"https://www.balbix.com/blog/glueball-cve-2020-
1464/"
                    ],
                    "other": [],
                    "patch": [

"https://portal.msrc.microsoft.com/en-US/security-
guidance/advisory/CVE-2020-1464"
                    ],
                    "workaround": []
                },
                "remediation": {
                    "solution": [
                        "Apply patch from
https://portal.msrc.microsoft.com/en-US/security-
guidance/advisory/CVE-2020-1464"
                    ],
                    "workaround": []
                },
                "risk": "Critical",
                "vulnerability_infection_methods":
[
                    "Attackers can load a malicious
file signed by a trusted developer and trick
Microsoft Windows into executing malicious code"
                ]
            }],
            "meta_info": {
                "creation_date": "29-11-2021
10:12:19",
                "last_modification_date": "29-11-
2021 10:12:19",
                "license": "Copyright (C) 2023
SecPod Technologies",
                "sp_ioc_id": "SP-IoC-000000",
                "sp_mve_id": "MVE-000360",
                "version": "2.0"
            },
            "threat_affected_cpes": {
                "cpe2.2": [
                    "cpe:/o:microsoft:windows_10:",
                    "cpe:/o:microsoft:windows_7:",

"cpe:/o:microsoft:windows_8.1:",

"cpe:/o:microsoft:windows_rt_8.1:",

"cpe:/o:microsoft:windows_server_2008:",

"cpe:/o:microsoft:windows_server_2012:",

"cpe:/o:microsoft:windows_server_2016:",

"cpe:/o:microsoft:windows_server_2019:"
                ],
                "cpe2.3": [

"cpe:2.3:o:microsoft:windows_10:",

"cpe:2.3:o:microsoft:windows_7:",

"cpe:2.3:o:microsoft:windows_8.1:",

"cpe:2.3:o:microsoft:windows_rt_8.1:",

"cpe:2.3:o:microsoft:windows_server_2008:",
```

Sample request	Sample response
	"cpe:2.3:o:microsoft:windows_server_2012:", "cpe:2.3:o:microsoft:windows_server_2016:", "cpe:2.3:o:microsoft:windows_server_2019:"] }, "threat_aliases": [], "threat_associated_c2_ips": [], "threat_associated_domains": [], "threat_capabilities": ["Arbitrary code can be executed with high privileges", "Malwares can be dropped into systems remotely"], "threat_cwe_ids": ["CWE-347"], "threat_description": "Glueball is a critical vulnerability arising due to a flaw where Microsoft Windows incorrectly validates file signatures for files being loaded. The flaw allows anyone to load improperly signed files and thus execute arbitrary code. This Issue was reported to Microsoft on 5th August 2018 but Microsoft provided a patch for this in the August 2020, only after the issue was being actively exploited in the wild.", "threat_exploited_cces": [], "threat_exploited_cisa_kevs": ["CVE-2020-1464"], "threat_exploited_cves": ["CVE-2020-1464"], "threat_extra_info": { "actively_exploited_cves": ["CVE-2020-1464"], "state_sponsored": "unknown", "unique_capabilities": [], "wildly_exploited": "yes" }, "threat_family": ["Exploit"], "threat_label": [], "threat_mitre_attack_info": { "data_sources": [], "from_mitre_attack": "yes", "group": {}, "mitigations": [], "references": [], "software": [], "tactics": [], "techniques": [] }, "threat_name": "Glueball", "threat_risk": "Critical" }] }] }

Possible Error Cases

- Invalid Organization Name

- Invalid Account Name
- Invalid User id.
- RP not enabled.

Patch Management

Saner Patch Management tool allows you to fetch all the security and non-security patches applicable to the devices in your organization and then perform remediation on these devices.

This section provides insights into the Saner Patch Management APIs and their usage.

Get all applicable Security Patches for remediation.

This method allows you to list all the missing security patches by group, hostname, or family for an account.

Method Name: **getApplicableRemediation**
Method Type: POST

Mandatory Parameters: accountid

Sample Request	Sample Response
<pre>{ "request": { "method": "getApplicableRemediation", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "Api_testing" }, { "key": "device", "values": ["desktop-jdn034t"] }] }]</pre>	<pre>{ "remediationinfo": [{ "account": "accc3", "assetpatchinfo": [{ "assetname": "Adobe Flash Player", "displayname": "Adobe Flash Player", "remjobpatchinfo": [{ "patchid": "flash_player-32.0.0.330-mac.pkg", "eri": "eri:com.secpod.eri:12092", "reboot": "FALSE", "fixinfo": "Security update available for Adobe Flash Player on Mac OS X" }</pre>

<table>
<tr>
<td><pre>
    }
  }
}
</pre></td>
<td><pre>
      ]
    }
  ]
 }
]
}
</pre></td>
</tr>
</table>

Possible Error Cases:

- Failed due to invalid account name or Id : <account>.
- Service PM is not enabled.
- No devices present for current selection.
- Invalid Input.
- No applicable missing patches/fixes found.

Add Security Remediation Job

This method allows you to create a remediation job for Missing security and non-security patches. The remediation job patch information will be retrieved from the “getApplicableRemediation” API.

Method Name: **createRemediationJob**
Method Type: POST

Mandatory Parameters: accountid, name, schedule, starttime, startwindowtime, device, and remjobpatchesinfo

<table>
<tr><th>Sample Request</th><th>Sample Response</th></tr>
<tr>
<td><pre>
{
 "request": {
  "method": "createRemediationJob",
  "parameters": {
   "parameterset": [
    {
     "parameter": [
      {
       "key": "accountid",
       "value": "Api_testing"
      },
      {
       "key": "name",
       "value": "MyJob1"
      },
      {
       "key": "schedule",
       "value": "2023-04-01"
      },
      {
       "key": "starttime",
       "value": "02:00:PM"
      },
      {
       "key": "startwindowtime",
       "value": "03:00:PM"
      },
      {
       "key": "startmessage",
       "value": "Remediationstarting"
      },
      {
       "key": "endmessage",
       "value": "RemediationFinished"
      },
      {
       "key": "autoreboot",
       "value": "FALSE"
      },
      {
</pre></td>
<td><pre>
{
 "response": {
   "method": "createRemediationJob",
       "results": {
        result": [{
         "key": "createRemediationJob",
              "status": "SUCCESS",
              "reason": ""
                     }]
              }
       }
}
</pre></td>
</tr>
</table>

<table>
<tr>
<td>

```
        "key": "rebootmessage",
        "value": "Rebootingin2minutes"
       },
       {
        "key": "forcereboot",
        "value": "FALSE"
       },
       {
        "key": "autoreboottime",
        "value": "2023-04-15.18:00:00"
       },
       {
        "key": "device",
        "values": [
         "desktop-jdn034t"
        ]
       },
       {
        "key": "remjobpatchesinfo",
        "valuesjson": [
         {
          "assetname": "MicrosoftWin-
dows1020h2x86",
          "remjobpatchinfo": [
           {
            "patchid": "Script_CVE-2013-
3900_fix.exe(script)",
            "eri": "eri:
com.secpod.eri:18033",
            "reboot": "TRUE"
           }
          ]
         },
         {
          "assetname": "MozillaFire-
foxx86",
          "remjobpatchinfo": [
           {
            "patchid": "Firefox-Setup-
98.0-x86.exe",
            "eri": "eri:
com.secpod.eri:21346",
            "reboot": "FALSE",
            "patch_grouptype": "SECU-
RITY"
           }
          ]
         }
        ]
       }
      ]
     }
    ]
   }
  }
}
```

</td>
<td></td>
</tr>
<tr>
<td>

For Non-Security Remediation Jobs

```
{
 "request": {
  "method": "createRemediationJob",
  "parameters": {
   "parameterset": [
    {
     "parameter": [
      {
       "key": "accountid",
       "value": "Demo Account"
      },
```

</td>
<td>

In case of success:

```
{
 "response": {
  "method": "createRemediationJob",
  "results": {
   "result": [
    {
     "key": "createRemediationJob",
     "status": "SUCCESS",
     "reason": ""
    }
   ]
  }
```

</td>
</tr>
</table>

```
        {
         "key": "name",
         "value": "Non Sec Rem Job"
        },
        {
         "key": "schedule",
         "value": "immediate"
        },
        {
         "key": "starttime",
         "value": ""
        },
        {
         "key": "startwindowtime",
         "value": ""
        },
        {
         "key": "startmessage",
         "value": "Remediationstarting"
        },
        {
         "key": "endmessage",
         "value": "RemediationFinished"
        },
        {
         "key": "autoreboot",
         "value": "FALSE"
        },
        {
         "key": "rebootmessage",
         "value": ""
        },
        {
         "key": "forcereboot",
         "value": "TRUE"
        },
        {
         "key": "autoreboottime",
         "value": "2025-04-15.18:00:00"
        },
        {
         "key": "device",
         "values": [
          "desktop-t2q9h33"
         ]
        },
        {
         "key": "remjobpatchesinfo",
         "valuesjson": [
          {
           "assetname": "7-zip x64",
           "remjobpatchinfo": [
            {
             "patchid": "7-zip-24.08-
x64.exe",
             "eri": "",
             "reboot": "",
             "patch_grouptype": "NON-SE-
CURITY"
            }
           ]
          }
         ]
        }
       ]
      }
     ]
    }
   }
  }
```

}	

Possible Error Cases

- Schedule value cannot be empty.
- Invalid schedule value
- Start time cannot be empty.
- Start window time cannot be empty.
- Invalid forcereboot value.
- Invalid start window time value.
- Invalid start time value.
- Invalid end time value.
- Start window time should be lesser than endtime.
- Start and end time cannot be same.
- By default, the patch_grouptype field takes SECURITY as the value. To create a Non-security remediation job, you must pass the value as 'NON-SECURITY' in the patch_grouptype field. The patch_grouptype field will take 'SECURITY' as its default value if no value is specified.

Get Job Status

This method allows you to get the remediation job status.

Method Name: getRemediationJobStatus

Method Type: POST

Mandatory Parameters: accountid and name

Sample Request

```
{
 "request": {
  "method": "getRemediationJobstatus",
  "parameters": {
   "parameterset": [
    {
     "parameter": [
      {
       "key": "accountid",
       "value": "{{account-name}}"
      },
      {
       "key": "name",
       "value": "{{job-Name}}"
      }
     ]
    }
   ]
  }
 }
}
```

Sample Response

```
{
"PatchingStatus": [{
"jobstatus": [{
"JobName": "MyJob14",
"Groupname": "windows10",
"Hostname": "User-laptop",
"Asset": "KB2267602",
"PatchName": "KB2267602",
"patchSize": "---",
"InstalledVersion": "---",
"Oldversion": "---",
"risksmitigated": "---",
"risksmitigatedcount": "---",
"Status": "fail",
"Reason": "Errorcode:0x80240022",
"LastUpdate": "2023-03-2809:25:10AMUTC"
}, {
"JobName": "MyJob14",
"Groupname": "windows10",
"Hostname": "desktop-q35u1cm",
"Asset": "KB2267602",
"PatchName": "KB2267602",
"patchSize": "---",
"InstalledVersion": "---",
"Oldversion": "---",
"risksmitigated": "---",
"risksmitigatedcount": "---",
"Status": "fail",
"Reason": "Errorcode:0x80240022",
"LastUpdate": "2023-03-2809:26:43AMUTC"
}, {
"JobName": "MyJob14",
"Groupname": "windows10",
"Hostname": "desktop-jdn034t",
```

	"Asset": "KB2267602", "PatchName": "KB2267602", "patchSize": "---", "InstalledVersion": "---", "Oldversion": "---", "risksmitigated": "---", "risksmitigatedcount": "---", "Status": "fail", "Reason": "Errorcode:0x80240022", "LastUpdate": "2023-03-2811:03:13AMUTC" }], "account": "Api_testing" }] }

Possible Error Cases:

- Failed due to invalid account name or Id : <account>.
- Failed due to invalid task name:<job name>.
- Invalid Input.
- Service PM is not enabled.
- Missing required parameters. Account name must be provided.
- Field <key> cannot be empty.
- No data found.
- Invalid device(s).
- Failed due to invalid name.

Get Remediation Job Details and Status

This method lets you get details of the remediation job for a set of devices and will return the response in a nested format. 'accountid' should be passed as part of the following query parameter: https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: getRemediationJob

Method Type: POST

Mandatory Parameters: jobname

Sample request	Sample response
{ "request": { "method": "getremediationjob", "parameters": { "parameterset": [{ "parameter": [{ "key": "jobname", "value": "myjob" }] }] } } }	In case of success: { "remediationJobDetails": { "devices-for-rem": { "device-for-rem": [{ "devicename": "192.168.XX.XX", "remediationjobstatus": { "reminfo": { "remname": "myremtask", "timestamp": "1438683587535", "remtimetype": "on-schedule", "remtime": "", "overallstatus": "fail", "patches": { "patch": [{ "status": "fail", "reason": "Failed, please rescan and retry remediation.", "appname": "linux-image-generic-3.19", "patchname": "" }] },

Sample request	Sample response


```
            "subsid": "bbbb-vbbb0eeerwerwer"
          }
        }
      },{
          "key": "startwindowtime",
          "value": "03:00:PM"
        },{
        "name": "abc-pc",
        "remediationjobstatus": {
          "reminfo": {
            "remname": " myremtask",
            "timestamp": "1438683587535",
            "remtimetype": "on-schedule",
            "remtime": "",
            "overallstatus": "success",
            "patches": {
              "patch": [{
                "status": "success",
                "reason": "",
                "appname": "GoogleChrome",
                "patchname": "chrome.exe"
              }]
            },
            "subsid": "aaaa-vbbb0eeerwerwer3"
          }
        }
      }]
    },
    "assets": {
      "asset": [{
        "displayname": "MozillaFirefox",
        "remediationPatches": {
          "remediationPatchInfo": [{
            "eriid": "eri: com.secpod.eri: 7486",
            "patchname": "Firefox-Setup-38.0-mac.dmg",
            "enabled": "true",
            "platform": "cpe: /o: apple: mac_os_x",
            "product": "cpe: /a: mozilla: firefox",
            "type": "VULNERABILITY"
          }]
        }
      },{
        "displayname": "AppleMacOSX10.9",
        "remediationPatches": {
        "remediationPatchInfo": [{
          "eriid": "eri: com.secpod.eri: 7438",
          "patchname": "cce-28301-0-patch.sh",
          "enabled": "true",
          "platform": "cpe: /o: apple: mac_os_x: 10.9",
          "product": "",
          "type": "COMPLIANCE"
        }]
      }]
    },
    "timestamp": "1437642107685",
    "desc": "Rem",
    "groupid": "[Arvindh-Systems]",
    "name": "All",
    "executiontype": "scheduled",
    "status": "completed"
  }
}
```


Possible Error Cases

- Invalid Input.
- Missing required parameters. Account name must be provided.

- Missing required parameters. Job name must be provided.

Get Remediation Patch Information

This method allows you to get details of the remediation patch to be applied for a set of devices. 'accountid' should be passed as part of the following query parameter:
https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: getRemediationJobPatch

Method Type: POST

Mandatory Parameters: patchname

Sample request	Sample response
<pre>{ "request": { "method": "getremediationjobpatch", "parameters": { "parameterset": [{ "parameter": [{ "key": "patchname", "value": "thunderbird.exe" }] }] } } }</pre>	In case of success: <pre>{ "patchName": "", "patchSize": "58MB", "rebootRequired": "FALSE", "desc": "Security update for Mozilla Products on Mac OS X", "instructions": "Install update 'Thunderbird-Setup-31.7.0-mac.dmg' on Mac OS X,", "prerequisites": "", "modifiedDate": "2018-06-03" }</pre>

Possible Error Cases

- Invalid Input.
- Field patchname cannot be empty.
- Service PM and CM are not enabled.
- Patch name cannot be empty.
- Failed due to invalid patch name.
- The patch <patch name> is not allowed to access. Service CM is not enabled.
- The patch <patch name> is not allowed to access. Service PM is not enabled.
- No data available.

Delete Remediation

This method is used to delete any security/non-security remediation job.

Method Name: deleteRemediation
Method Type: POST

Mandatory Parameters: accountid and name

Sample request	Sample response
<pre>{ "request": { "method": "deleteremediation", "parameters": { "parameterset": [{</pre>	<pre>{ "response": { "method": "deleteRemediation", "results": { "result": [{</pre>

Sample request	Sample response
<pre>"parameter": [{ "key": "accountid", "value": "secpod" }, { "key": "name", "value": "testfromUD_Compliance " }] }] } } }</pre>	<pre>"key": "testfromUD_Compliance", "status": "SUCCESS", "reason": "" }] } } }</pre>

Possible Error Cases

- Invalid Input
- Job Deletion Failed
- Failed due to invalid task name.
- Service PM is not enabled.
- Service CM is not enabled.
- Failed due to invalid account name or id.

Get All Applicable Non-Security Patches For Remediation

This method will list all the missing non-security patches by group, hostname, or family for an account.

Method Name: **getApplicableNonSecurityRemediation**
Method Type: POST

Mandatory Parameters: accountid

Sample Request	Sample Response
<pre>{ "request": { "method": "getapplicablenonsecurityremediation", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "Api_testing" }] }] } } }</pre>	<pre>{ "remediationinfo": [{ "account": "_Default.Product- Demo", "patchinfo": [{ "patchid": "KB5022478", "fixinfo": "2023-01 Cumulative Update Preview for .NET Framework 3.5, 4.8 and 4.8.1 for Windows 10 Version 22H2 for x64 (KB5022478)", "reboot": "TRUE" }, { "patchid": "KB2267602", "fixinfo": "Security Intelligence Up-date for Microsoft Defender Antivirus - KB2267602 (Version 1.381.2886.0)", "reboot": "FALSE" }], "patchname": ["KB5022478", "KB2267602"] }</pre>

	<pre>] }</pre>

Possible Error Cases:

- Failed due to invalid account name or Id : <account>.
- Service PM is not enabled.
- No applicable devices found.
- Invalid Input.

Get Non-Security Job Details

This method fetches the details of a non-security remediation , firmware, rollback and reboot jobs in Saner PM. Details such as 'schedule', 'start time', 'end time', 'startwindowtime', 'remtype', 'startmessage', and 'endmessage' are fetched.

You need to provide a unique job name while creating an IR job in Saner PM. You can't give the job name as PM or CM.

Method Name: getNonSecJobDetails

Method Type: POST

Mandatory Parameters: name and accountid

Sample request	Sample response
<pre>{ "request": { "method": "getnonsecjobdetails", "parameters": { "parameterset": [{ "parameter": [{ "key": "name", "value": "non_sec_job" }, { "key": "accountid", "value": "Test_Account" }] }] } } }</pre>	<pre>For IR job { "IRDetails": [{ "appandpatchmgmt": { "patchmgmt": [{ "patchid": "URL=https://192.168.2.23/AncorWebService/sanergetresource ?resource=7-zip-23.01-x86", "command": "install", "silentoption": "/S", "type": "MISSINGPATCHES", "asset_name": "7-zip x86" }], "appmgmt": [], "starttime": "01:00:AM", "startwindowtime": "02:00:AM", "endtime": "08:00:AM", "autoreboot": "true", "forcereboot": "false", "rebootdatetime": "2024-01-25 09:00:AM", "schedule": "2024-01-25", "type": "remediationjob", "remtype": "VULNERABILITY", "rebootmessage": "TEST" }, "account": "Test_Account" }] }</pre> For Rollback { "IRDetails": [

Sample request	Sample response
	{ "rollback": [{ "name": "KB5027538", "assetname": "KB5027538", "type": "original" }], "schedule": "on-schedule", "autoreboot": "false", "forcereboot": "false", "account": "Test_Account" }] }

Possible Error Cases

- Invalid Input.
- Failed due to invalid account name or id.
- User/Account need subscription to perform operation.
- Failed due to invalid task name.

Get Applicable Firmware Remediation

This method gets the list of firmware patches available for the account or the list of devices provided.

Method Name: getAppplicableFirmwareRemediation
Method Type: POST

Mandatory Parameters: accountid and group

Sample request	Sample response
{ "request": { "method": "getapplicablefirmwareremediation", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "TestAccount" }, { "key": "device", "values": ["desktop-rg93vgt"] }] }] } } }	{ "remediationinfo": [{ "account": "TestAccount", "patchinfo": [{ "patchname": "ATA Channel 0", "patchid": "pid:PCIIDE\\IDECHANNEL\\4&3084357F&0&0", "detected": "2024-08-20 01:42:03 PM UTC", "vendor": "microsoft" }, { "patchname": "ATA Channel 1", "patchid": "pid:PCIIDE\\IDECHANNEL\\4&3084357F&0&1", "detected": "2024-08-20 01:42:03 PM UTC", "vendor": "microsoft" }, { "patchname": "Common KVM processor", "patchid": "pid:ACPI\\GENUINEINTEL_- _INTEL64_FAMILY_15_MODEL_6_- _COMMON_KVM_PROCESSOR_3",

Sample request	Sample response
	"detected": "2024-08-20 01:42:03 PM UTC", "vendor": "microsoft" }], "patchname": ["pid:PCIIDE\\IDECHANNEL\\4&3084357F&0&0", "pid:PCIIDE\\IDECHANNEL\\4&3084357F&0&1", "pid:ACPI\\GENUINEINTEL_-_INTEL64_FAMILY_15_MODEL_6_-_COMMON_KVM_PROCESSOR_3"] }] }

Possible Error Cases

- Invalid input.
- Failed due to invalid account name or id.
- No applicable devices found.
- No applicable missing patches/fixes found.

Create Firmware Remediation Job

This method allows users to create remediation jobs for firmware patches. 'accountid' should be passed as part of the following query parameter:
https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: **createFirmwareRemediationJob**

Method Type: POST

Mandatory Parameters: accountid, name, schedule, starttime, startwindowtime, device, and patchname.

Sample request	Sample response
{ "request": { "method": "createFirmwareRemediationJob", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "_Default" }, { "key": "name", "value": "Test_Job" }, { "key": "schedule", "value": "2024-12-30" }, { "key": "starttime", "value": "03:20:PM" },	{ "response": { "method": "createFirmwareRemediationJob", "results": { "result": [{ "key": "createRemediationJob", "status": "SUCCESS", "reason": "" }] } } }

Sample request	Sample response
{ "key": "startwindowtime", "value": "04:00:PM" }, { "key": "endtime", "value": "05:00:PM" }, { "key": "forcereboot", "value": "false" }, { "key": "device", "values": ["desktop-test"] }, { "key": "remjobpatchesinfo", "patchinfos": [{ "patchname": "ATA Channel 1", "patchid": "pid:PCIIDE\\IDECHANNEL\\4&3084357F&0&1", "detected": "2024-08-20 01:42:03 PM UTC", "vendor": "microsoft" }, { "patchname": "Common KVM processor", "patchid": "pid:ACPI\\GENUINEINTEL_- _INTEL64_FAMILY_15_MODEL_6_- _COMMON_KVM_PROCESSOR_3", "detected": "2024-08-20 01:42:03 PM UTC", "vendor": "microsoft" }] }] }] } } }	

Possible Error Cases

- Invalid start window time value.
- Invalid starttime value.
- Invalid endtime value.
- Start window time should be lesser than end time.
- Start and end time cannot be same.

Get Firmware Remediation Job Status

This method gets the status of the job/task created as part of the firmware remediation job.

Method Name: getFirmwareRemediationJobStatus

Method Type: POST

Mandatory Parameters: accountid and name

Sample request	Sample response
{ "request": { "method": "getfirmwareremediationjobstatus", "parameters": { "parameterset": [{ "parameter": [{ "key": "name", "value": "firmware dioublepatchsecpod" }, { "key": "accountid", "value": "checking" }, { "key": "device", "values": ["secpod"] }, { "key": "group", "values": ["windows 11"] }, { "key": "family", "values": ["windows"] }] }] } } }	{ "response": { "method": "getFirmwareRemediationjobStatus", "results": { "result": [{ "key": "getFirmwareRemediationjobStatus", "status": "FAIL", "reason": "Failed due to invalid account name or Id: checkingdj:';" }] } } }

Possible Error Cases

- Invalid task name.
- Invalid account name or id.
- Invalid input.

Delete Firmware Remediation Job

This method is used to delete any existing firmware remediation jobs.

Method Name: deleteFirmwareRemediationJob

Method Type: POST

Mandatory Parameters: accountid and name

Sample request	Sample response
<pre>{ "request": { "method": "deletefirmwareremediationjob", "parameters": { "parameterset": [{ "parameter": [{ "key": "name", "value": "firmtest" }, { "key": "accountid", "value": "TestAccount" }] }] } } }</pre>	<pre>{ "response": { "method": "deleteFirmwareRemediation", "results": { "result": [{ "key": "firmtest", "status": "SUCCESS", "reason": "" }] } } }</pre>

Possible Error Cases

- Invalid Input
- Failed due to invalid task name.
- Failed due to invalid account name or id.

Get All Applicable Remediation Rules based on Groups

This method allows you to get all applicable remediation rules for one or multiple groups. 'accountid' should be passed as part of the following query parameter:
https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: getAllApplicableRules

Method Type: POST

Mandatory Parameters: groupname

Sample request	Sample response
<pre>{ "request": { "method": "getallapplicablerules", "parameters": { "parameterset": [{ "parameter": [{ "key": "groupname", "value": "debian" }, { "key": "apifilters", "filters": [{ "tags": {</pre>	<pre>In case of success: { "ruleRemediationJobDetails": { "name": "rule test", "desc": "this is an example of rule", "autoremediate": "OFF", "timestamp": "1434269311107", "assets": { "asset": [{ "name": "Mozilla Firefox x86", "enabled": "" },{ "name": "libssl1.0.0", "enabled": "" },{ "name": "VBScript 5.7",</pre>

```
          "in": {
           "owner": [
            "User_1"
           ]
          },
          "not in": {
           "owner": [
            "User_1"
           ]
          }
         }
        }
       ]
      }
     ]
    }
   ]
  }
 }
}
```

```
        "enabled": ""
      },{
        "name": "OpenSSL x86",
        "enabled": ""
      },{
        "name": "VBScript 5.8",
        "enabled": ""
      }]
    },
    "groups": {
      "group": [{
        "groupid": "centos"
      }]
    }
  }
}
```


Possible Error Cases

- Invalid Input.
- No Records.
- Unable to fetch rule details.

Get Remediation Rule

This method allows you to get details of remediation rule for a set of devices.
'accountid' should be passed as part of the following query parameter: https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: getRemediationRule

Method Type: POST

Mandatory Parameters: rulename

Sample request	Sample response
{ "request": { "method": "getremediationrule", "parameters": { "parameterset": [{ "parameter": [{ "key": "rulename", "value": "rulewin8-8.1" }] }] } } }	In case of success: { "ruleRemediationJobDetails": { "name": "rule test", "desc": "this is an example of rule", "timestamp": "1434269311107", "assets": { "asset": [{ "name": "Mozilla Firefox x86", "enabled": "" },{ "name": "libssl1.0.0", "enabled": "" },{ "name": "VBScript 5.7", "enabled": "" },{ "key": "startwindowtime", "value": "03:00:PM" },

Sample request	Sample response
	<pre>{ "name": "OpenSSL x86", "enabled": "" },{ "name": "VBScript 5.8", "enabled": "" }] }, "groups": { "group": [{ "groupid": "centos" }] } } }</pre>

Possible Error Cases

- Invalid Input
- Failed due to invalid rule name.
- No Records
- Unable to fetch rule details.

Add Remediation Rule

This method allows you to add remediation rule for a set of groups with the collection of applicable products and platforms from 'getallapplicablerules' request. Request 'getapplicableremediation' is mandatory before creating remediation rule. Required fields are *rulename* and *ruledesc*. These fields should only contain alphanumeric characters, space, dot(.), underscore(_) or hyphen. The included assets will always be taken into consideration for remediation process that occurs after scan. The excluded list of products and platforms will not be remediated.

Value of *type* field could be "vulnerability" or "compliance".

Note: It is not mandatory to provide a value for 'remediationdelayindays'. If no value is provided, by default, 0 is taken as default value for 'remediationdelayindays'.

Method Name: addRemediationRule

Method Type: POST

Mandatory Parameters: rulename, type, starttime, startwindowtime,scheduletype, groupname, includeasset

Sample request	Sample response
<pre>{ "request": { "method": "addRemediationRule", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "test_account" }, { "key": "rulename",</pre>	In case of success: <pre>{ "response": { "method": "addremediationrule ", "results": { "result": [{ "key": "tagsRule", "status": "SUCCESS", "reason": "" }] } } }</pre>

Sample request	Sample response
<pre>"value": "tagsRule" }, { "key": "ruledesc", "value": "rule with tags" }, { "key": "type", "value": "VULNERABILITY" }, { "key": "starttime", "value": "03:24:PM" }, { "key": "endtime", "value": "03:50:PM" }, { "key": "startwindowtime", "value": "03:30:PM" }, { "key": "scheduletype", "value": "daily" }, { "key": "groupname", "value": "debian" }, { "key": "forcereboot", "value": "true" }, { "key": "remediationdelayindays", "value": "5" }, { "key": "includeasset", "value": "apparmor" }, { "key": "skipfeatureupdate", "value": "true" }, { "key": "apifilters", "filters": [{ "tags": { "in": { "Owner": ["User1", "User2"] }, "not in": { "Category": ["Finance"] } } } }] }]</pre>	In case of failure: <pre>{ "response": { "method": "addremediationrule", "results": { "result": [{ "key": "tagsRule", "status": "FAIL", "reason": "Invalid Input" }] } } }</pre>

Sample request	Sample response
}] } } } _____	

Possible Error Cases

- Invalid Input.
- Failed due to invalid rule name.
- Failed due to invalid rule description.
- Rule creation Failed.
- Failed due to duplicate rule name.
- Invalid type.
- Field type cannot be empty.
- Field scheduletype cannot be empty.
- Start time cannot be empty.
- Start window time cannot be empty.
- Invalid forcereboot value.
- Invalid start window time value.
- Invalid starttime value.
- Invalid endtime value.
- Start window time should be lesser than end time.
- Start and end time cannot be same.
- The range of allowed remediation delay is 1 to 7 days.

Update Remediation Rule

This method allows you to update remediation rule for a set of groups. Required field is *rulename*. The included assets will always be taken into consideration for remediation process that occurs after scan. The excluded list of products and platforms will not be remediated. Value of type could be "vulnerability" or "compliance".

Note: It is required to provide a complete list of exclude or include assets during update along with the changes required. However, it is not mandatory to provide a value for 'remediationdelayindays'. If no value is provided, by default, 0 is taken as default value for remediationdelayindays key.

Method Name: updateRemediationRule

Method Type: POST

Mandatory Parameters: rulename, type, starttime, startwindowtime, scheduletype, and groupname

Sample request	Sample response
{ "request": { "method": "updateRemediationRule", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid",	In case of success: { "response": { "method": "updateremediationrule",

Sample request	Sample response
<pre>"value": "spXXXX" }, { "key": "rulename", "value": "tagRule" }, { "key": "ruledesc", "value": "rule with tags" }, { "key": "type", "value": "VULNERABILITY" }, { "key": "starttime", "value": "03:24:PM" }, { "key": "endtime", "value": "03:50:PM" }, { "key": "startwindowtime", "value": "03:30:PM" }, { "key": "scheduletype", "value": "daily" }, { "key": "groupname", "value": "debian" }, { "key": "forcereboot", "value": "true" }, { "key": "remediationdelayindays", "value": "5" }, { "key": "includeasset", "value": "apparmor" }, { "key": "skipfeatureupdate", "value": "true" }, { "key": "apifilters", "filters": [{ "tags": { "in": { "Owner": ["User_1", "User_2"] }, "not in": { "Category": ["Finance"] } }</pre>	<pre>"results": { "result": [{ "key": "tagRule", "status": "SUCCESS", "reason": "" }] } } }</pre> In case of failure: <pre>{ "response": { "method": "updateremediationrule", "results": { "result": [{ "key": "tagRule", "status": "FAIL", "reason": "Invalid Input" }] } } }</pre>

Sample request	Sample response
}] }] }] } } }	
{ "request": { "method": "updateremediationrule ", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "secpod" }, { "key": "rulename", "value": "rulewin8-8.1" }, { "key": "ruledesc", "value": "rule for all windows 8 and 8.1 groups" }, { "key": "groupname", "value": "windows 8" }, { "key": "groupname", "value": "windows 8.1" }, { "key": "excludeasset", "value": "Microsoft .NET Framework 4.5 SP1" }, { "key": "excludeasset", "value": "Apple Safari x86" }, { "key": "includeasset", "value": "Adobe Reader 11 x86" }, { "key": "includeasset", "value": "Microsoft Windows 8" }, { "key": "scheduletype", "value": "2023-08-01" }, { "key": "forcereboot", "value": "FALSE" }, { "key": "startwindowtime", "value": "03:00:PM" }, { "key": "excludeasset", "value": "Microsoft Windows 8.1"	

Sample request	Sample response
<pre>}, { "key": "remediationdelayindays", "value": "6" }, { "key": "skipfeatureupdate", "value": "true" }, { "key": "type", "value": "vulnerability" }]}]}}}</pre>	

Possible Error Cases

- Invalid Input
- Failed due to invalid rule name.
- Failed due to invalid rule description.
- Rule update Failed.
- Provide valid severity value.
- Invalid type.
- Field type cannot be empty.
- Start time cannot be empty.
- Start window time cannot be empty.
- Invalid forcereboot value.
- Invalid start window time value.
- Invalid starttime value.
- Invalid end time value.
- Start window time should be lesser than end time.
- Start end time cannot be same.
- The range of allowed remediation delay is 1 to 7 days.
- Group(s) not present.

Delete Remediation Rule

This method allows you delete remediation rule for a set of groups. 'accountid' should be passed as part of the following query parameter: https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: deleteRemediationRule

Method Type: POST

Mandatory Parameters: rulename

Sample request	Sample response
<pre>{ "request": { "method": "deleteremediationrule", "parameters": { "parameterset": [{ "parameter": [{ "key": "rulename", "value": "rulewin8-8.1"</pre>	<pre>In case of success: { "response": { "method": "deleteremediationrule", "results": { "result": [{ "key": "rulewin8-8.1", "status": "SUCCESS", "reason": "" }]</pre>

}] }] } } }	} } }
	In case of failure: { "response": { "method": "deleteremediationrule", "results": { "result": [{ "key": "rulewin8-8.1", "status": "FAIL", "reason": "Failed due to invalid rule name" }] } } }

Possible Error Cases

- Invalid Input.
- Failed due to invalid rule name.
- Rule deletion Failed.

Get Automation Rule Status

This method returns whether the automation rule is paused or running. If the rule name is not specified, the API will return the results for all the rules.

.

Method Name: getAutomationRuleStatus

Method Type: POST

Mandatory Parameters: accountname

Sample request	Sample response
{ "request": { "method": "getAutomationRuleStatus", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountname", "value": "Test_Account" }, { "key": "rulenames", "values": ["automationTest","test rule"] }] }] } } }	In case of success: { "automationRuleStatusInfo": [{ "rule_name": "automationTest", "rule_status": "running" }, { "rule_name": "test rule", "rule_status": "paused" }] }

Possible Error Cases

- Missing mandatory fields. Account name must be provided
- Field key cannot be empty
- Invalid rule name
- Failed due to invalid account name or Id
- Field key is invalid
- Account name must be provided
- Remediation rules {rule names} do not exist
- Failed to get automation rule status.

Update Automation Rule Status

This method allows you to pause or resume a remediation rule in Saner PM (Patch Management). The key "status" accepts 'resume' and 'pause' as values.

.

Method Name: updateAutomationRuleStatus

Method Type: POST

Mandatory Parameters: accountname, rulenames, and status

Sample request	Sample response
<pre>{ "request": { "method": "updateAutomationRuleStatus", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountname", "value": "Test_Account" }, { "key": "rulenames", "values": ["automationTest", "automationTest2"] }, { "key": "status", "value": "pause" }] }] } } }</pre>	<pre>In case of success: { "response": { "method": "updateAutomationRuleStatus", "results": { "result": [{ "key": "updateAutomationRUleStatus", "status": "success", "reason": "Updated remediation automation rules successfully, one or more rules were already in paused state" }] } } }</pre>
<pre>---------- To resume an automation rule:</pre>	<pre>---------- In case of success: {</pre>

Sample request	Sample response
<pre>{ "request": { "method": "updateAutomationRuleStatus", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountname", "value": "Test_Account" }, { "key": "rulenames", "values": ["automationTest"] }, { "key": "status", "value": "resume" }] }] } } }</pre>	<pre>"response": { "method": "updateAutomationRuleStatus", "results": { "result": [{ "key": "updateAutomationRuleStatus", "status": "success", "reason": "Updated remediation automation rule successfully" }] } } }</pre>

Possible Error Cases

- Missing mandatory fields. Account name, rule name and status must be provided
- Field key cannot be empty
- Invalid rule name {name}
- Invalid status
- Invalid value. Field status can be either pause or resume
- Failed due to invalid account name or Id
- Field key is invalid
- Account name must be provided
- Status must be provided
- Rule names must be provided
- Remediation rules {rule names} do not exist
- Failed to update automation rule

Get Applicable Rollback Patches

You can get a list of all installed patches applicable for the rollback using this method.

Method Name: getPatchesForRollback

Method Type: POST

Mandatory Parameters: accountid

Sample Request	Sample Response
<pre>{ "request": { "method": "getPatchesForRollback",</pre>	<pre>{ "remediationinfo": [{</pre>

```
  "parameters": {
   "parameterset": [
    {
     "parameter": [
      {
       "key": "accountid",
       "value": "Api_testing"
      }
     ]
    }
   ]
  }
 }
}
```

```
   "account": "ACC_22Dec",
   "devicerollbackpatchinfo": [
    {
     "devicename": "desktop-d04sj8i",
     "rollbackpatchinfo": [
      {
       "patchid": "KB5021089",
       "asset": "2022-12 Cumulative
Update for .NET Framework 3.5, 4.8 and
4.8.1 for Windows 10 Version 22H2 for
x64 (KB5021089)",
       "size": "67.7 MiB",
       "rollbackstatus": "TRUE",
       "installeddate": "2022-12-23
07:01:57 AM UTC"
      },
      {
       "patchid": "KB5012170",
       "asset": "2022-12 Security
Update for Windows 10 Version 22H2 for
x64-based Sys-tems (KB5012170)",
       "size": "116.2 KiB",
       "rollbackstatus": "TRUE",
       "installeddate": "2022-12-23
07:01:57 AM UTC"
      }
     ]
    },
    {
     "devicename": "win-o5bmvc1lpno",
     "rollbackpatchinfo": [
      {
       "patchid": "KB2898871",
       "asset": "Security Update for
Mi-crosoft .NET Framework 4.5.1 on
Windows 8.1 and Windows Server 2012 R2
for x64-based Systems (KB2898871)",
       "size": "35.3 MiB",
       "rollbackstatus": "TRUE",
       "installeddate": "2022-12-23
08:21:47 AM UTC"
      }
     ]
    }
   ]
  }
 ]
}
```


Possible Error Cases:

- Failed due to invalid account name or Id : <account>.
- Service PM is not enabled.
- No applicable devices found.
- Invalid Input.
- No Records.
- Device does not exist
- Field ‘device’ cannot be empty.

Rollback Job

This method allows you to roll back the installed patches for a group. ‘accountid’ should be passed as part of the following query parameter:
https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: **createPatchRollbackTask**

Method Type: POST

Mandatory Parameters: accountid, name, schedule, starttime, startwindowtime, and rollbackpatchesinfo

Sample Request

```
{
 "request": {
  "method": "createPatchRollbackTask",
  "parameters": {
   "parameterset": [
    {
     "parameter": [
      {
       "key": "accountid",
       "value": "Api_testing"
      },
      {
       "key": "name",
       "value": "MyRollbackJob1"
      },
      {
       "key": "schedule",
       "value": "2023-04-01"
      },
      {
       "key": "starttime",
       "value": "15:24:00"
      },
      {
       "key": "startmessage",
       "value": "Remediationstarting"
      },
      {
       "key": "endmessage",
       "value": "RemediationFinished"
      },
      {
       "key": "autoreboot",
       "value": "TRUE"
      },
      {
       "key": "rebootmessage",
       "value": "Rebootingin2minutes"
      },
      {
       "key": "startwindowtime",
       "value": "03:00:PM"
      },
      {
       "key": "forcereboot",
       "value": "FALSE"
      },
      {
       "key": "autoreboottime",
       "value": "2023-04-15.18:00:00"
      },
      {
       "key": "rollbackpatchesinfo",
       "valuesjson": [
        {
         "devicename": "desktop-
q35u1cm",
         "rollbackpatchinfo": [
          {
           "patchid": "KB5009467"
          },
          {
           "patchid": "KB5001716"
          }
```

Sample Response

```
{
 "response": {
  "method": "createPatchRollbackTask",
  "results": {
   "result": [
    {
     "key": "createPatchRollbackTask",
     "status": "SUCCESS",
     "reason": ""
    }
   ]
  }
 }
}
```

<pre>] }] }] }] } }</pre>	

Possible Error Cases

- Schedule value cannot be empty.
- Invalid schedule value
- Start time cannot be empty.
- Start window time cannot be empty.
- Invalid forcereboot value.
- Invalid start window time value.
- Invalid starttime value.
- Invalid endtime value.
- Start window time should be lesser than end time.
- Start and end time cannot be same.

Get Patch Rollback Status

This method is used to fetch the status of job/task and related info created as part of patch rollback status.

Method Name: getPatchRollbackStatus

Method Type: POST

Mandatory Parameters: accountid and group

Sample request	Sample response
<pre>{ "request": { "method": "getpatchrollbackstatus", "parameters": { "parameterset": [{ "parameter": [{ "key": "name", "value": "TestRollbackStatusAPI" }, { "key": "accountid", "value": "{{AccountName}}" }, { "key": "device", "values": ["{{hostname}}"] }, { "key": "group", "values": ["oracle linux"</pre>	<pre>{ "RollbackStatus": [{ "accountrollbackstatus": [{ "JobName": "TestRollbackStatusAPI", "CreationTime": "2023-08-22 09:38:05 AM UTC", "Groupname": "oracle linux", "Hostname": "qa-oracle-linux-8.3-x64", "LastUpdate": "2023-08-22 09:39:09 AM UTC", "ActualDate": "Saner Agent: 2023-08-22 09:39:09 AM UTC", "jobType": "IR", "PatchName": "abattis-cantarell-fonts", "Status": "fail", "Reason": "An unknown error has occurred while un-installing the patch." }], "account": "AlltoolsEnabled" }] }</pre>

Sample request	Sample response
<pre>] }, { "key": "family", "values": ["unix"] }] }] } } }</pre>	

Possible Error Cases

- Invalid Input.
- Failed due to invalid account name or Id : <account>.
- Invalid Input: <job name>.

Delete Patch Rollback Task

This method is used to delete any created patch rollback task.

Method Name: deletePatchRollbackTask

Method Type: POST

Mandatory Parameters: accountid and name

Sample request	Sample response
<pre>{ "request": { "method": "deletepatchrollbacktask", "parameters": { "parameterset": [{ "parameter": [{ "key": "name", "value": "TestRollbackStatusAPI" }, { "key": "accountid", "value": "{{AccountName}}" }] }] } } }</pre>	<pre>{ "response": { "method": "deletePatchRollbackTask", "results": { "result": [{ "key": "rollback", "status": "SUCCESS", "reason": "" }] } } }</pre>

Possible Error Cases

- Invalid Input.
- Operation unsuccessful.
- Failed due to invalid task name.
- Failed due to invalid account name or id.

Reboot Device

This method allows you to reboot your device. You can reboot one or more hosts and hosts belonging to a group simultaneously.

Method Name: rebootDevice

Method Type: POST

Mandatory Parameters: All the fields specified in the request are mandatory.

Sample Request	Sample Response
<pre>{ "request": { "method": "rebootdevice", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "Api_testing" }, { "key": "starttype", "value": "immediate" }, { "key": "rebootmessage", "value": "Your system will be rebooted shortly" }, { "key": "name", "value": "rebootsingledevice" }, { "key": "device", "values": ["desktopq35u1cm"] }] }] }}}</pre>	<pre>{ "response": { "method": "rebootdevice", "results": { "result": [{ "key": "1", "status": "SUCCESS", "reason": "addedreboottasksuccessfully" }] } } }</pre>

Possible Error Cases:

- Invalid Input.
- Didn't get accountname or no valid startype found.
- invalid account name.
- Invalid taskname.
- Invalid task id name.
- accountid not found.
- Service PM is not enabled.
- already registered response name.
- invalid starttime.
- invalid endtime.
- invalid date.
- invalid device name(s).
- Invalid input for device/group.
- Reboot failed from PM command side.

Reboot Task Status

You can use this method to check the status of reboot tasks.

Method Name: getRebootTaskStatus

Method Type: POST

Mandatory Parameters: accountid and name

Sample Request	Sample Response
<pre>{ "request": { "method": "getreboottaskstatus", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "Api_testing" }, { "key": "name", "value": "reboot-group-sched" }] }] } } }</pre>	<pre>{ "Result": [{ "jobname": "reboot-group-sched", "status": [{ "hostname": "windows-laptop", "overallstatus": "Waitingfordevicetorespond", "lastupdate": "" }, { "hostname": "desktop-jdn034t", "overallstatus": "Waitingfordevicetorespond", "lastupdate": "" }, { "hostname": "desktopq35u1cm", "overallstatus": "Received", "lastupdate": "1648721865898" }], "accountid": "Api_testing" }]</pre>

Possible Error Cases:

- Invalid Input.
- invalid account name.
- Invalid taskname.
- accountid not found.
- Task doesn't exist.
- Task status returned an invalid response.

Delete Reboot Task

This method allows you to delete the reboot task of your account.

Method Name: deleteRebootTask

Method Type: POST

Mandatory Parameters: accountid and name.

Sample Request	Sample Response
<pre>{ "request": { "method": "deletereboottask", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "Api_testing" },</pre>	<pre>{ "response": { "method": "Response", "results": { "result": [{ "key": "1", "status": "Success", "reason": "Successfullydeletedtask" }</pre>

```
        {
         "key": "name",
         "value": "reboot-task"
        }
       ]
      }
     ]
    }
   }
}
```

```
    ]
   }
  }
}
```

Possible Error Cases:

- Invalid Input
- Invalid account name
- Invalid taskname
- accountid not found
- Service PM is not enabled
- task doesn't exist
- Failed to delete task

Exclude a Patch or Asset

This API allows administrators to exclude either a specific patch or an entire asset/application from visibility and remediation based on the defined scope. For example, a single patch for an application can be excluded, or the application itself can be excluded across assets. Exclusions can be applied at the account, group, or device level and are automatically removed once the configured exclusion period expires.

Method Name: **excludevulnerability**

Method Type: POST

Mandatory Parameters: account_name, policy_name, item_type, items, scopeType, selectedScopeItems, module_type

To Exclude a Patch :

Sample Request

```
{
  "request": {
    "method": "excludevulnerability",
    "parameters": {
      "parameterset": [
        {
          "parameter": [
            {
                "key":"account_name",
                "value": "Saner-Account"
            },
            {
              "key": "policy_name",
              "value": "policy001"
            },
            {
              "key": "item_type",
              "value": "PATCH"
            },
            {
              "key": "items",
```

Sample Response

In case of success:

```
{
  "response": {
    "method": "excludevulnerability",
      "results": {
        "result": [{
           "key": " policy001",
           "status": "SUCCESS",
           "reason": "Success!"
                }]
        }
    }
}
```

In case of failure:

```
{
  "response": {
    "method": "excludevulnerability",
      "results": {
```

```
                "values": [
                  "flash-player-
32.0.0.445_x64-rpm.rpm"
                ]
              },
              {
                "key": "days",
                "value": "1"
              },
              {
                "key": "reason",
                "value": "RISK_ACCEPTED"
              },
              {
                "key": "desc",
                "value": ""
              },
              {
                "key": "comments",
                "value": ""
              },
              {
                "key": "scopeType",
                "value": "group"
              },
              {
                "key": "selectedScopeItems",
                "values": [
                  "debian12"
                ]
              },
              {
                  "key": "module_type",
                  "value": "PM"
              }
            ]
          }
        ]
      }
    }
}
```

```
          "result": [{
             "key": " policy001",
             "status": "FAIL",
             "reason": "Duplicate exclude
policy name"
                  }]
          }
      }
}
```

To Exclude an Asset :

Sample Request

```
{
  "request": {
    "method": "excludevulnerability",
    "parameters": {
      "parameterset": [
        {
          "parameter": [
            {
                "key":"account_name",
                "value": "Saner-Account"
            },
            {
              "key": "policy_name",
              "value": " policy001"
            },
            {
              "key": "item_type",
              "value": "ASSET"
```

Sample Response

In case of success:

```
{
  "response": {
    "method": "excludevulnerability",
     "results": {
       "result": [{
         "key": " policy001",
         "status": "SUCCESS",
         "reason": "Success!"
                }]
          }
      }
}
```

In case of failure:

```
{
```

<table>
<tr>
<td>

```
            },
            {
              "key": "items",
              "values": [
                "7-zip x64"
              ]
            },
            {
              "key": "days",
              "value": "1"
            },
            {
              "key": "reason",
              "value": "RISK_ACCEPTED"
            },
            {
              "key": "desc",
              "value": ""
            },
            {
              "key": "comments",
              "value": ""
            },
            {
              "key": "scopeType",
              "value": "Devices"
            },
            {
              "key": "selectedScopeItems",
              "values": [
                "win10-106",
                "win10-115"
              ]},
            {
                "key": "module_type",
                "value": "PM"
            }]
        }]
    }
  }
}
```

</td>
<td>

```
  "response": {
  "method": "excludevulnerability",
  "results": {
   "result": [{
      "key": " policy001",
      "status": "FAIL",
      "reason": "Duplicate exclude policy
name"
                }]
        }
    }
}
```

</td>
</tr>
</table>

Endpoint Management

Saner Endpoint Management tool allows you to install and uninstall software on your endpoint devices. At the same time, you can perform automated software deployments using the Saner Endpoint Management tool.

This section gives insights into all the Saner Endpoint Management APIs and their usage.

Add Software Deployment Job

This method creates a software deployment job. Users can add a URL-encoded string to the 'filename' field. Deployment jobs can be created for multiple devices of the same group.

Method Name: addSoftwareDeployment

Method Type: POST

Mandatory Parameters: name, accountid, filename, devicename, and scheduletype

Sample request	Sample response
<pre>{ "request": { "method": "addsoftwaredeployment", "parameters": { "parameterset": [{ "parameter": [{ "key": "name", "value": "myfirstsoftwaredeploy" },{ "key": "filename", "value": "winscp-5.13.4-setup-x86" },{ "key": "starttime", "value": "06:00:AM" },{ "key": "endtime", "value": "06:20:AM" },{ "key": "scheduletype", "value": "date|immediate|on-schedule" },{ "key": "schedulevalue", "value": "2019-06-21" },{ "key": "persistent", "value": "no" },{ "key": "periodicity", "value": "0" },{ "key": "autoreboot", "value": "true" },{ "key": "rebootdatetime", "value": "2019-06-29 00:00:00" },{ "key": "groupname", "value": "windows 10" },{ "key": "devicename", "value": "my-pc" },{ "key": "startwindowtime", "value": "03:00:PM" },{ "key": "forcereboot", "value": "FALSE" },{"key": "accountid", "value": "testaccount1"</pre>	<pre>In case of success: { "response": { "method": "addsoftwaredeployment", "results": { "result": [{ "key": "", "status": "SUCCESS", "reason": "" }] } } }</pre>

Sample request	Sample response
}]}]}}}	

Possible Error Cases

- Invalid Account
- Invalid Input
- Duplicate job name
- File not found.
- Invalid Encoded Value
- Schedule value cannot be empty.
- Schedule type cannot be empty.
- Invalid schedule type.
- Invalid schedule value
- Starttime cannot be empty.
- Start window time cannot be empty.
- Invalid forcereboot value.
- Invalid start window time value.
- Invalid starttime value.
- Invalid endtime value.
- Start window time should be lesser than end time.
- Start and end time cannot be same.

Add Software Provision

This method allows users to create a software deployment rule. When Saner Agent is installed on an end-point, all the software mentioned in this rule gets automatically deployed on the endpoint.

Method Name: addSoftwareProvision

Method Type: POST

Mandatory Parameters: name, filename, accountid, and rebootdatetime(this parameter is required if the user needs to provide a custom rebootdatetime.)

Sample request	Sample response
{ "request": { "method": "addsoftwareprovision", "parameters": { "parameterset": [{ "parameter": [{ "key": "name", "value": "test" }, { "key": "filename", "value": "npp.8.5.6.Installer.x64" }, { "key": "autoreboot", "value": "true" }, { "key": "rebootdatetime", "value": "06:06:AM"	In case of success: { "response": { "method": "addsoftwareprovision", "results": { "result": [{ "key": "", "status": "SUCCESS", "reason": "" }] } } }

Sample request	Sample response
<pre>}, { "key": "groupname", "value": "debian" }, { "key": "accountid", "value": "spXXXX" }, { "key": "apifilters", "filters": [{ "tags": { "in": { "Owner": ["User 1", "User 2"] }, "not in": { "Category": ["Finance"] } } }] }] }] } }</pre>	

Possible Error Cases

- Invalid Account ID.
- Invalid Input.
- Duplicate rule name.
- File not found.
- Invalid Encoded Value.

Upload URL for Software Deployment

Users can use this method to upload a URL directly to the sever and install later. The 'filename' field which accepts a URL in an encoded format.

Method Name: uploadSoftwareUrl

Method Type: POST

Mandatory Parameters: filename, name, and accountid

Sample request	Sample response
<pre>{ "request": { "method": "uploadsoftwareurl",</pre>	<pre>In case of success: {</pre>

Sample request	Sample response
<pre>"parameters": { "parameterset": [{ "parameter": [{ "key": "filename", "value":"https://sbp.enterprisedb.com/getfile.jsp?fileid=1258649 },{ "key": "name", "value": "PostgreSQL" },{ "key": "family", "value": "windows" },{ "key": "version", "value": "1.0" },{ "key": "architecture", "value": "x86" },{ "key": "category", "value": "Application Software" },{ "key": "installoption", "value": "-s" },{ "key": "publisher", "value": "PostgreSQL Global Development Group" },{ "key": "description", "value": "PostgreSQL Open RDBMS" },{ "key": "accountid", "value": "testaccount1" }] }] } }}</pre>	<pre>"response": { "method": "uploadsoftwareurl", "results": { "result": [{ "key": "", "status": "SUCCESS", "reason": "" }] } } }</pre>

Possible Error Cases

- Invalid Account.
- Invalid Input.
- Invalid URL Encoded Value.

Upload Installer Package for Software Deployment

This method allows you to upload different executable files such as .exe, .dmg, Linux binaries etc. Users can provide URL encoded value in the 'filename' field. The field 'data' will only accept Base64 encoded content of the installer binary.

Following is a python snippet to read an installer binary:

```
import base64

with open("test.exe", "rb") as f:
    encodedZip = base64.b64encode(f.read())
```

Method Name: uploadInstallerPackage

Method Type: POST

Mandatory Parameters: data, filename, and accountid.

Sample request	Sample response
{ "request": { "method": "uploadinstallerpackage", "parameters": { "parameterset": [{ "parameter": [{ "key": "data", "value": " FuIGlzIGR…" },{ "key": "name", "value": "WinSCP" },{ "key": "filename", "value": "WinSCP-5.13.4-Setup.exe" },{ "key": "family", "value": "windows" },{ "key": "version", "value": "1.0" },{ "key": "architecture", "value": "x86" },{ "key": "category", "value": "Application Software" },{ "key": "installoption", "value": "-s" },{ "key": "publisher", "value": " Martin Prikryl" },{ "key": "description", "value": "Transfer files with WinSCP" },{ "key": "accountid", "value": "testaccount1" }] }] } } }	In case of success: { "response": { "method": "uploadinstallerpackage", "results": { "result": [{ "key": "", "status": "SUCCESS", "reason": "" }] } } }

Possible Error Cases

- Invalid Account
- Invalid Input
- Invalid Encoded Value
- Invalid family
- Unsupported file type. Supported types are : deb | dmg | elf | exe | msi | pkg | rpm | rar

Upload Compressed File for Software Deployment

This method allows users to upload compressed files like in .zip format. Users can provide URL encode value in the 'filename' field. The field 'data' will only accept Base64 encoded content of the compressed installer file.

Following is a Python snippet to read an installer compressed file,

```
import base64
with open("test.zip", "rb") as f:
    encodedZip = base64.b64encode(f.read())
```

Method Name: uploadCompressedFile

Method Type: POST

Mandatory Parameters: data, filename, and accountid.

Sample request	Sample response
{ "request": { "method": "uploadcompressedfile", "parameters": { "parameterset": [{ "parameter": [{ "key": "data", "value": "UEsDBAoAAAAAABw+ElcAAAAAAAAAAAAAAAAFAB- wAYS50eHRVVAkAA2ci32RnIt9kdXgLAAEE- AAAAAAQAAAAAUEsBAh4DCgAAAAAAHD4SVwAAAAAAAAAAAAAAAAUA- GAAAAAAAAAAAAKSBAAAAAGEudHh0VVQFAANnIt9kdXgLAAEE- AAAAAAQAAAAAUEsFBgAAAAABAAEASwAAAD8AAAAAAA==" },{ "key": "filename", "value": "abc.zip" },{ "key": "name", "value": "Test User" },{ "key": "family", "value": "windows" },{ "key": "version", "value": "1.0" },{ "key": "architecture", "value": "x86" },{ "key": "category", "value": "Application Software" },{ "key": "runfile", "value": "/" },{ "key": "publisher", "value": "testpub" },{ "key": "extractlocation", "value": "/" },{ "key": "description", "value": "PuTTY SSH key generation utility" },{ "key": "installoption", "value": "/S" },{ "key": "accountid", "value": "Test_User" }] }] }}}	In case of success: { "response": { "method": "uploadcompressedfile", "results": { "result": [{ "key": "", "status": "SUCCESS", "reason": "" }] } } }

Possible Error Cases

- Invalid Account ID.
- Invalid Input.
- Invalid Encoded Value.
- Invalid family.
- Unsupported file type. Only zip | gzip | tar are allowed.

Create Uninstall Task

This method allows users to create tasks to uninstall an application. Users can provide URL encoded value in the 'appname' field.

Method Name: uninstallSoftware

Method Type: POST

Mandatory Parameters: name, appname, accountid, devicename, and scheduletype.

Sample request

```
{
  "request": {
    "method": "uninstallsoftware",
    "parameters": {
      "parameterset": [{
        "parameter": [{
          "key": "name",
          "value": "RemovePuttyJob"
        },{
          "key": "appname",
          "value": "PuTTY"
        },{
          "key": "starttime",
          "value": "06:00:AM"
        },{
          "key": "endtime",
          "value": "06:20:AM"
        },{
          "key": "scheduletype",
          "value": "date"
        },{
          "key": "schedulevalue",
          "value": "2019-06-21"
        },{
          "key": "persistent",
          "value": "no"
        },{
          "key": "periodicity",
          "value": "0"
        },{
          "key": "autoreboot",
          "value": "true"
        },{
          "key": "rebootdatetime",
          "value": "2019-06-29.00:00:00"
        },{
          "key": "silentoption",
          "value": "/s"
        },{
          "key": "groupname",
          "value": "windows 10"
        },{
          "key": "devicename",
          "value": "devname"
        },{
          "key": "startwindowtime",
          "value": "06:10:AM"
        },{
          "key": "forcereboot",
           "value": "FALSE"
        },{
          "key": "accountid",
```

Sample response

In case of success:

```
{
  "response": {
    "method": "uninstallsoftware",
    "results": {
      "result": [{
        "key": "",
        "status": "SUCCESS",
        "reason": ""
      }]
    }
  }
}
```

Sample request	Sample response
<pre>"value": "testaccount1" }] }] } } }</pre>	

Possible Error Cases

- Invalid Account.
- Invalid Input.
- Invalid Encoded Value.
- Invalid family.
- Schedule value cannot be empty.
- Schedule type cannot be empty.
- Invalid schedule value.
- Start time cannot be empty.
- Start window time cannot be empty.
- Invalid forcereboot value.
- Invalid start window time value.
- Invalid starttime value.
- Invalid endtime value.
- Start window time should be lesser than end time.
- Start and end time cannot be same.

Get All Applications

This method will give the details of all available applications that can be used by users to create a job or to set a software provision rule.

Method Name: getAllApplications

Method Type: POST

Mandatory Parameters: accountid

Sample request	Sample response
<pre>{ "request": { "method": "getallapplications", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "testaccount1" }] }] } } }</pre>	<pre>In case of success: { "response": { "method": "getallapplications", "results": { "result": [{ "key": "", "status": "success", "reason": "", "assetsinfo": [{ "assetname": "EaSynth ForeUI", "family": "windows", "category": "Application Software", "versions": [{ "filename": "ForeUI", "version": "", "architecture": "x86", "type": "exe", "family": "windows",</pre>

Sample request	Sample response
	<pre>"size": "136.0 KiB" }] }] }] } } }</pre>

Possible Error Cases

- Invalid Account ID.

Reports Management

Saner provides multiple reports that give you visibility into your Organization's Assets, Vulnerabilities, and Device Details.

This section gives insights into all the Saner Report APIs and their usage.

Get Compliance Profile Evaluation

This method allows you get evaluated compliance profile report of all devices corresponding to a profile. 'accountid' should be passed as part of the following query parameter:
https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: getProfileReport

Method Type: POST

Mandatory Parameters: profilename

Sample request	Sample response
<pre>{ "request": { "method": "getprofilereport", "parameters": { "parameterset": [{ "parameter": [{ "key": "profilename", "value": "myprofile" }] }] } } }</pre>	<pre>{ "profilereport": { "device": [{ "name": "acer-p645-mg-pc", "categories": [{ "category": { "name": "A.5.1.1 Policies for information security", "rules": [{ "rule": { "name": "Rule for A.5.1.1 Policies for information security", "status": "PASS" } }] } },{ "category": { "name": "A.5.1.2 Review of the policies for information security", "rules": [{ "rule": { "name": "Rule for A.5.1.2 Review of the policies for information security", "status": "FAIL" } }] } },{ "category": { "name": "A.6.1.1 Information security roles and responsibilities", "rules": [{ "rule": {</pre>

Sample request	Sample response
	<pre>"name": "Rule for A.6.1.1 Information security roles and responsibilities", "status": "NOT_CHECKED" } }] } }] }] }}</pre>

Possible Error Cases

- No Records if no such profile or group exists.

Get All Vulnerable Assets

This method allows you to get all vulnerable asset names installed in the organization. These vulnerable asset names can be used at input for creating remediation jobs. 'accountid' should be passed as part of the following query parameter: https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: getVulnerableAssets

Method Type: POST

Mandatory Parameters: No mandatory parameters needed.

Sample request	Sample response
<pre>{ "request": { "method": "getvulnerableassets", "parameters": {} } }</pre>	<pre>{ "asset": [{ "name": "Microsoft Office Access 2007" },{ "name": "OpenSSL x86" }] }</pre>

Possible Error Cases

- No Records.

Get All Assets

This method allows you to get all software asset names installed in the organization. 'accountid' should be passed as part of the following query parameter: https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: getAssets

Method Type: POST

Mandatory Parameters: No mandatory parameters needed.

Sample request	Sample response
<pre>{ "request": { "method": "getassets", "parameters": {} } }</pre>	<pre>{ "asset": [{ "name": "libssl0.9.8" },{ "name": "libjavascriptcoregtk-3.0-0" },{ "name": "Microsoft Office Access 2007" },{ "name": "OpenSSL x86" },{ "name": "libxvmc1" },{ "name": "7-Technologies Interactive Graphical SCADA System" }] }</pre>

Possible Error Cases

- No Records.

Get Assets By Vulnerability

This method is used to get the lists of assets linked to the CVE or CVEs specified in the reference value.

Method Name: getAssetsByVulnerability

Method Type: POST

Mandatory Parameters: accountid and reference

Sample request	Sample response
<pre>{ "request": { "method": "getassetsbyvulnerability", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "TestAccount" }, { "key": "reference", "values": ["CVE-2021-3331", "CVE-2022-32168"] }] }] }</pre>	<pre>{ "vulnerableassetsinfo": [{ "assetname": "Notepad++ plus x64", "title": "DLL hijacking vulnerability in Notepad++ - CVE-2022-32168", "reference": "CVE-2022-32168", "detectiondate": "2023-08-08", "hostname": "desktop-rg93vgt", "operatingsystem": "Microsoft Windows 10 v22H2 architecture 64-bit", "group": "windows 10" }, { "assetname": "WinSCP x86", "title": "Remote code execution vulnerability in WinSCP - CVE-2021-3331", "reference": "CVE-2021-3331", "detectiondate": "2023-08-08",</pre>

Sample request	Sample response
} }	"hostname": "desktop-rg93vgt", "operatingsystem": "Microsoft Windows 10 v22H2 architecture 64-bit", "group": "windows 10" }] }

Possible Error Cases

- Invalid Input
- Invalid Account
- Invalid Reference
- Not enough inputs found.

Get All Profile names

This method allows you to get all the Benchmark Profile Names that exist in an account. And once you retrieve these profile names, you can assign them to groups. 'accountid' should be passed as part of the following query parameter: https://saner.secpod.com/AncorWebService/perform?accountid=<accountname>

Method Name: getProfileNames

Method Type: POST

Mandatory Parameters: No parameters needed.

Sample request	Sample response
{ "request": { "method": "getprofilenames", "parameters": {} } }	{ "profile": [{ "name": "win8profile" },{ "name": "win7profile" },{ "name": "macprofile" },{ "name": "testprofile" }] }

Possible Error Cases

- No Records.

Get All Saved Report Names

This method allows you to view all the saved report names, including canned and custom reports.

Method Name: getReportNames

Method Type: POST

Mandatory Parameters: accountid

Sample request	Sample response
<pre>{ "request": { "method": "getreportnames", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "testaccount1" }] }] } } }</pre>	<pre>{ "reportnames": ["Asset Report", "Compliance Report", "Endpoint Management Report", "Executive Report", "Patch Report", "Patching Impact Report", "Posture Anomaly Report", "Risk Assessment Report", "Risk Prioritization Report", "Vulnerability Report", "TestReports"] }</pre>

Possible Error Cases

- Invalid Account

Get All Report Names of Each Tool

This method allows you to view reports for each of the following service-enabled tools.

Asset Exposure(AE)

Patch Management (PM)

Risk Prioritization (RP)

Compliance Management (CM)

Vulnerability Management (VM)

Posture Anomaly (PA) & Endpoint Management (EM).

Also,this API allows you to fetch reports for Cyber Hygiene Score (CHS).

Method Name: getReportApis

Method Type: POST

Mandatory Parameters: accountid

Sample request	Sample response
<pre>{ "request": {</pre>	<pre>{ "reportsapis": [{</pre>

Sample request	Sample response
<pre>"method": "getreportapis", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "testaccount1" }] }] } } }</pre>	<pre>"tool": "AE", "names": ["Applications added to Blacklist", "Applications added to White-list", "Application Details", "Assets Based On Publisher", "Blacklisted Applications", "Currently Monitored applications", "Device Details", "Device Manufacturer", "Hardware Asset", "Hardware Licenses", "Operating System Licenses", "Outdated Applications", "Outdated OS", "Overall License violations", "Rarely Used Applications", "Software Asset Summary", "Software Licenses", "Unauthorized Software Applications", "Unauthorized Software Applications Count"] }] }</pre>

Possible Error Cases

- Invalid Account

Get Report API Data

This method allows you to view the data for each report based on the specified account and tool.

Method Name: getReportApiData

Method Type: POST

Mandatory Parameters: accountid, reportapi, organization, and tool

Sample request	Sample response
<pre>{ "request": { "method": "getreportapidata", "parameters": { "parameterset": [{ "parameter": [{ "key": "organization", "value": "TempOrg2" }, {</pre>	<pre>{ "referencesview": [{ "reference": "CVE-2013-3869", "title": "Denial of service vulnerability in Digital Signatures in Microsoft Windows", "highFidelityAttacks": [], "severityscore": "5.0", "severity": "Medium", "totalassets": 1, "assets": ["Microsoft Windows Server 2012 R2 x64"</pre>

```
        "key": "accountid",
        "value": "acc7582"
       },
       {
        "key": "reportapi",
        "value": "All
Vulnerabilities"
       },
       {
        "key": "tool",
        "value": "VM"
       },
       {
        "key": "startdate",
        "value": "2021-08-17"
       },
       {
        "key": "enddate",
        "value": "2021-09-16"
       },
       {
        "key": "sanerenabledhosts",
        "value": "true"
       },
       {
        "key": "apifilters",
        "filters": [
         {
          "tags": {
           "in": {
            "Owner": [
             "User_1",
             "User_2"
            ]
           },
           "not in": {
            "Category": [
             "Finance"
            ]
           }
          }
         }
        ]
       }
      ]
     }
    ]
   }
  }
}
```

```
    ],
    "hosts": 1,
    "hostnames": [
     "win-o5bmvc1lpno"
    ],
    "detectiondate": "2022-12-22",
    "releasedate": "2013-11-13",
    "fixinfo": "This is Microsoft OS or product
remediation. Saner will install all latest updates
for this product silently. It is recommended that you
do not power off machine or disconnect network when
this is going on."
   },
   {
    "reference": "CVE-2013-2566",
    "title": "Plaintext recovery vulnerability in RC4
algorithm in Web Browsers via statistical analysis of
ciphertext",
    "highFidelityAttacks": [],
    "severityscore": "5.9",
    "severity": "Medium",
    "totalassets": 1,
    "assets": [
     "Microsoft Internet Explorer 11"
    ],
    "hosts": 1,
    "hostnames": [
     "win-o5bmvc1lpno"
    ],
    "detectiondate": "2022-12-22",
    "releasedate": "2013-03-15",
    "fixinfo": ""
   }
  ]
}
```


Possible Error Cases

- Report not found.
- Invalid Account.
- Invalid key found.
- Service not found.
- Invalid tool Name
- Invalid input found for agent-enabled hosts.
- Service not enabled.
- Tag value <tag_value> of key <tag_key> is invalid. Value should be either true or false.

- Tag operation cannot be empty.
- Invalid tag operation: <operation_name>
- Tag key cannot be empty.
- Tag values are empty for tag key <tag_key>.
- Invalid tag key <tag_key>.
- Tag value is empty for tag key <tag_key>.
- Invalid tag value <tag_value> for tag key <tag_key>.
- Invalid installed start date
- Invalid installed end date
- Invalid released start date
- Invalid released end date
- Installed start date should not be greater than current date
- Invalid installed start date. Date should be in yyyy-MM-dd format
- Installed end date should not be greater than current date
- Invalid installed end date. Date should be in yyyy-MM-dd format
- Released start date should not be greater than current date
- Invalid released start date. Date should be in yyyy-MM-dd format
- Released end date should not be greater than current date
- Invalid released end date. Date should be in yyyy-MM-dd format
- Installed end date should be between installed start date and current date
- Released end date should be between released start date and current date

The “getreportapidata” API supports the following fields for specific reports:

1.“startdate”

2.“enddate”

3.“sanerenabledhosts”

4.“organization”.

“startdate” and “enddate” (optional): These two fields are introduced to provide a date range which will help in getting Weekly, Monthly, Quarterly reports. The format for the start and end date should be “yyyy-mm-dd”. If this field is empty, the report will include all the information irrespective of the date. The following reports supports ‘startdate’ and ‘enddate’.

- Remediation Patch Details By Asset Name.
- Misconfiguration Fix Details By Asset Name.
- Remediation Patch Details By Task Name.
- Misconfiguration Fix Details By Task Name.
- Remediation Patch Details By Group Name.
- Misconfiguration Fix Details By Group Name.
- Remediation Patch Details By Host Name.
- Misconfiguration Fix Details By Host Name.
- Patch Remediation Summary.
- Misconfiguration Fix Summary.
- Patch Job Status
- Job Status Summary

- Patch Rollback Job Status
- Test and Deploy Patch Job Status
- Remediation Rule Status
- Installed Patches
- Installed Patches by Devices?
- Misconfiguration Fixes Job Status
- Rollback Misconfiguration Fixes Job Status
- Firmware Job Status
- Patch Job Status
- Job Status Summary
- Patch Rollback Job Status
- Test and Deploy Patch Job Status
- Remediation Rule Status
- Installed Patches
- Installed Patches by Devices?
- Misconfiguration Fixes Job Status
- Rollback Misconfiguration Fixes Job Status
- Firmware Job Status
- All Vulnerabilities Based on Asset
- Vulnerabilities by Devices

“sanerenabledhosts” (optional): By default, remediation details reports will include saner-enabled and disabled devices. Users can get remediation details only for the Saner-enabled devices by settingthe input value as “true”.

“organization” (mandatory): To get reports at the organization level that includes all the accounts, provide the organization name, and do not set accountid. To get reports for a particular account in an organization, you need to set both "organization" and "accountid" key fields.

Key ‘reportapifilters’ has been added to the ’getreportapidata’ You can now filter based on reference (CVE ID), severity, hosts, families, application, applicationExclude, patchType and limit search results.

Four keys namely –‘installedstartdate’, ‘installedenddate’, ‘releasedstartdate’, ‘releasedenddate’ have been added to the below reports.

- Patch Compliance by Devices.
- Installed Patches by Devices.
- Missing Patches by Devices.

'installedstartdate' and 'installedenddate' : These two keys accept the start date and end date during which the patches were installed.

'releasedstartdate' and 'releasedenddate' : These two keys accept the start date and end date during which the patches were released.

Sample Rest API Request and Response: Sample JSON for the "Patch Compliance By Devices"report.

Sample request

```
{
"request": {
"method": "getReportApiData",
"parameters": {
"parameterset": [
{
"parameter": [
{
"key": "organization",
"value": "Test Org"
},
{
"key": "accountid",
"value": "Test Account"
},
{
"key": "reportapi",
"value": "Patch Compliance by Devices"
},
{
"key": "tool",
"value": "PM"
},
{
"key": "installedstartdate",
"value": "2022-09-12"
},
{
"key": "installedenddate",
"value": "2024-03-06"
},
{
"key": "releasedstartdate",
"value": "2022-09-12",
{
"key": "releasedenddate",
"value": "2024-03-06"
}]}]}}}
```

Sample response

```
{
 "patchDetails": [
  {
   "hostName": "test_machine",
   "ipAddress": "192.168.X.X",
   "operatingSystem": "Ubuntu v18.04 ar-
chitecture x86_64",
   "groupName": "ubuntu",
   "assetName": "libdrm2",
   "patchName": "libdrm-common",
   "patchVersion": "2.4.101-2~18.04.1",
   "releaseDate": "2023-02-08",
   "installedDate": "2024-03-05",
   "installedBy": "SecPod Saner",
   "status": "Installed"
  },
  {
   "hostName": "test machine",
   "ipAddress": "192.168.X.X",
   "operatingSystem": "Ubuntu v18.04 ar-
chitecture x86_64",
   "groupName": "ubuntu",
   "assetName": "libdrm2",
   "patchName": "libdrm2:amd64",
   "patchVersion": "2.4.101-2~18.04.1",
   "releaseDate": "2023-02-08",
   "installedDate": "2024-03-05",
   "installedBy": "SecPod Saner",
   "status": "Installed"
  }
 ]
}
```

Sample Rest API Request and Response: Sample JSON for the "Remediation Patch Details By Host Name"report.

Method name: getreportapidata

Sample request	Sample response
{ "request": { "method": "getreportapidata", "parameters": { "parameterset ": [{ "parameter" : [{ "key": "organization", "value": "testorganization1" },{ "key": "accountid", "value": "testaccount1" },{ "key": "reportapi", "value": "Remediation PatchDetails By Host Name" },{ "key": "tool", "value": "PM" },{ "key": "startdate", "value": "2023-08-17" },{ "key": "enddate", "value": "2024-09-16" },{ "key": "sanerenabledhosts", "value": "true" }] }] } } }	{ "remediationdetailsbyhos t": [{"hostname": "abc- pc", "taskscount": 4, "task names": "pm_rule, fri011, pm_job01,two","assetscount": 5, "assetnames": "accountsservice, libbluetooth3, aspell,apparmor,avahi- daemon", "fixedpatchescount ": 5, "fixedpatches": "libaccountsservice0: amd64, libbluetooth3:amd64,libaspell15: amd64,apparmor,avahi-daemon", "failedpatchescount": 1, "failedpatches": "busybox- initramfs", "risksmitigatedcount": 9, "criticalriskscount": 2, "criticalrisks": "CVE-2019-17544,CVE-2017- 6519", "highriskscount": 4, "highrisks": "CVE-2020-0556,CVE-2020- 27153,CVE-2021-3468,USN-3784-1", "mediumriskscount": 2, "mediumrisks": "CVE-2020-26558,CVE-2018-14036", "lowriskscount": 1, "lowrisks": "CVE-2020- 16126", "totalrisksbeforepatch": 2493, "totalrisksafterpatch": 2484,"account": "testaccount" }] }

Sample Rest API Request and Response: Sample JSON for the “Misconfiguration Fix Summary” report

Method name: getreportapidata

Sample request	Sample response
{ "request": { "method": "getreportapidata", "paraeters": { "parameterset": [{ "parameter": [{ "key": "organization", "value": "testorganization1" },{ "key": "accountid", "value": "testaccount1" },{ "key": "reportapi", "value": "Misconfiguration Fix Summary"	{ "taskcount": 5, "assetcount": 7, "patchcount": 11, "riskcount": 13, "hostcount": 2 }

<table>
<tr>
<td>

```
        },{
          "key": "tool",
          "value": "VM|AM|PM|CM|PA|EM"
        },{
          "key": "startdate",
          "value": "yyyy-MM-dd"
        },{
          "key": "enddate",
          "value": "yyyy-MM-dd"
        },{
          "key": "sanerenabledhosts",
          "value": "true"
        }]
      }]
    }
  }
    }
```

</td>
<td></td>
</tr>
</table>

A new key namely – ‘createdBy’ has been added to the below reports. Also, the below reports support the ‘startdate’ and ‘enddate’ keys.

- Patch Job Status
- Job Status Summary
- Patch Rollback Job Status
- Test and Deploy Patch Job Status
- Remediation Rule Status
- Installed Patches
- Installed Patches by Devices?
- Misconfiguration Fixes Job Status
- Rollback Misconfiguration Fixes Job Status
- Firmware Job Status

Sample Rest API Request and Response: Sample JSON for the “Patch Job Status” report

Method name: getreportapidata

<table>
<tr><th>Sample request</th><th>Sample response</th></tr>
<tr>
<td>

```
{  {
 "request": {
  "method": "getreportapidata",
  "parameters": {
   "parameterset": [
    {
     "parameter": [
      {
       "key": "accountid",
       "value": "Demo Account"
      },
      {
       "key": "organization",
       "value": "Documentation"
      },
      {
       "key": "reportapi",
       "value": "Patch Job Status"
      },
```

</td>
<td>

```
{
 "jobstatus": [
  {
   "JobName": "Test_Patch",
   "CreationTime": "2024-11-28 05:09:29
AM UTC",
   "Groupname": "Custom Group",
   "Hostname": "520-av-windows11",
   "Asset": "Adobe Reader 10.1 x86",
   "PatchName": "acrobat-reader-dc-con-
tinuous-24.003.20112-win32.exe",
   "patchSize": "456872570",
   "InstalledVersion": "24.003.20112",
   "Oldversion": "Unknown",
   "risksmitigated": [
    "CVE-2015-4444",
    "CVE-2015-4445",
    "CVE-2012-4147",
    "CVE-2015-4446",
```

</td>
</tr>
</table>

```
        {
         "key": "tool",
         "value": "PM"
        },
        {
         "key": "startdate",
         "value": "2024-11-01"
        },
        {
         "key": "enddate",
         "value": "2024-11-30"
        },
        {
         "key": "reportapifilters",
         "filters": [
          {
           "createdBy": [
            "Admin"
           ]
          }
         ]
        }
       ]
      }
     ]
    }
   }}
```

```
     "CVE-2015-4447",
     "CVE-2015-4448",
     "CVE-2012-4149",
     "CVE-2015-4449",
     "CVE-2012-4148",
     "CVE-2013-3341"
    ],
    "risksmitigatedcount": 10,
    "Status": "success",
    "Reason": "The remediation task in-
stalled successfully.",
    "LastUpdate": "2024-11-28 05:22:20 AM
UTC",
    "CreatedBy": "Admin",
    "jobType": "Rem Job"
   }
  ]
}
```


Pagination Support

To improve performance and scalability when retrieving large datasets, pagination support has been introduced for selected reports under the Report API (getReportAPIData).
Instead of returning all results in a single response, the API returns data in manageable pages, enabling efficient data retrieval for reports with a large number of records.

The **getReportApiData** API supports pagination through standard pagination parameters that control page size, navigation, and result metadata. These parameters allow clients to request specific pages of data and determine whether additional pages are available.

Supported Pagination Parameters

The following parameters are available in the API response to support pagination:

- **limit** – Number of results requested per page
- **currentPage** – Current page number in the result set
- **pageSize** – Number of results returned in the current page
- **totalEntries** – Total number of records available for the report
- **isNextPage** – Indicates whether a next page of data is available
- **isPrevPage** – Indicates whether a previous page of data is available
- **totalPages** – Total number of pages generated based on the specified limit

Reports Supporting Pagination

Pagination is currently enabled for the following report

- **DM** - All Devices, Device Details Summary
- **AE** - Assets Based On Publisher, Operating System Licenses, Software Licenses, Hardware Licenses, Outdated Applications, Blacklisted Applications, Rarely Used Applications, Device Details, Application Details, Software Asset Summary, Application By Devices
- **VM** - High Fidelity Attacks, All Vulnerabilities, Vulnerability Count Based On Operating System, Vulnerability Count Based On Group, Vulnerabilities by Devices, All Vulnerabilities Based On Asset
- **EM** - All Devices, Not Scanned Devices, Device Details Summary

- **PM** - Missing Patches, Missing Patches by Devices, Assets With No Patches, Missing Security Patches for Outdated Products, Outdated Asset Patches, Outdated OS Patches, Devices with Missing Security Patches, Top 10 Missing Security Patches, Top 10 Devices By Security Patches, Firmware, Installed Patches, Installed Patches by Devices, Installed Patches by Devices, Remediation Patch Details By Task Name (Job), Third-Party Security Patches
- **CM** - Misconfigurations, Top 10 Recommended CCE Remediation

Sample Request and Response:

Sample request

```
{
 "request": {
   "method": "getReportApi-
Data",
    "parameters": {
     "parameterset": [{
     "parameter": [{
       "key": "organization",
    "value": "Saner-Org" },
                 {
    "key": "accountid",
    "value": "Saner-Account"
    },
                 {
    "key": "reportapi",
"value":    "All Vulnerabili-
ties" },
                 {
    "key": "tool", "value":
    "VM" },
                 {
    "key": "limit", "value":
    2},
                 {
    "key": "page", "value":
    1},
                 {
      "key": "apifilters",
      "filters": [{
        "patchType": ["os"],
        "families": ["unix",
    "windows","macos"],
       "application":["bind"],
       "applicationEx-
    clude":true
                    }
                  ]
                }
              ]
            }
          ]
        }
      }
    }
```

Sample response

```
    {
        "referencesview": [
            {
                "exploitability": "No",
                "reference": "CVE-2025-9900",
                "detectiondate": "2026-01-09",
                "releasedate": "2025-09-23",
                "highFidelityAttacks": [],
                "highFidelityAttacksCount": 0,
                "zeroDayVulnerability": "FALSE",
                "assetFamily": [
                    "unix"
                ],
                "severity": "High",
                "severityscore": 8.8,
                "patchType": "vendor",
                "platforms": [
                    "cpe:/o:ubuntu:ubuntu_linux:20.04"
                ],
                "products": [
                    "cpe:/a:libtiff:libtiff:5"
                ],
                "assets": [
                    "libtiff5"
                ],
                "vulnerableInstanceCount": 1,
                "family": "unix",
                "product": "cpe:/a:libtiff:libtiff:5",
                "platform": "cpe:/o:ubuntu:ub-
untu_linux:20.04",
                "hosts": 1,
                "hostnames": [
                    "ubuntu-20.04-x64"
                ],
                "totalassets": 1,
                "title": "Write-What-Where Vulnerabil-
ity in Libtiff",
                "fixinfo": "This is an Operating Sys-
tem package remediation. Saner will install all
latest updates for this product silently.\nIt is
recommended that you do not power off machine/dis-
connect network when this is going
on.<br><br><strong>Patch
Name(s):</strong><ul><li>libtiff5</li></ul><br>Ap-
plying these patch(s) will address the related se-
curity vulnerability.",
```

```
                "patch": "libtiff5",
                "patchcount": 1,
                "portservice": "-"
            },
            {
                "exploitability": "No",
                "reference": "CVE-2025-9714",
                "detectiondate": "2026-01-09",
                "releasedate": "2025-09-10",
                "highFidelityAttacks": [],
                "highFidelityAttacksCount": 0,
                "zeroDayVulnerability": "FALSE",
                "assetFamily": [
                    "unix"
                ],
                "severity": "Medium",
                "severityscore": 5.5,
                "patchType": "vendor",
                "platforms": [
                    "cpe:/o:ubuntu:ubuntu_linux:20.04"
                ],
                "products": [
                    "cpe:/a:xmlsoft:libxml2"
                ],
                "assets": [
                    "libxml2"
                ],
                "vulnerableInstanceCount": 1,
                "family": "unix",
                "product": "cpe:/a:xmlsoft:libxml2",
                "platform": "cpe:/o:ubuntu:ub-
untu_linux:20.04",
                "hosts": 1,
                "hostnames": [
                    "ubuntu-20.04-x64"
                ],
                "totalassets": 1,
                "title": "Uncontrolled Recursion Vul-
nerability in libxml2 for XPath Evaluation",
                "fixinfo": "This is an Operating Sys-
tem package remediation. Saner will install all
latest updates for this product silently.\nIt is
recommended that you do not power off machine/dis-
connect network when this is going
on.<br><br><strong>Patch
Name(s):</strong><ul><li>libxml2</li></ul><br>Ap-
plying these patch(s) will address the related se-
curity vulnerability.",
                "patch": "libxml2",
                "patchcount": 1,
                "portservice": "-"
            }
        ],
        "currentPage": 1,
        "pageSize": 2,
        "totalEntries": 818,
        "isNextPage": true,
```

	<pre>"isPrevPage": false, "totalPages": 409 }</pre>

Get PDF of Saved Report

Users can download all reports data of canned or custom report. This method will provide the response as a zip file which contains a pdf file.

Method Name: **getPdfReport**

Method Type: POST

Mandatory Parameters: accountid and reportname

Sample request	Sample response
<pre>{ "request": { "method": "getpdfreport", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "testaccount1" },{ "key": "reportname", "value": "testreport1" }] }] }}}</pre>	<pre>Response will be available in pdf file.</pre>

Possible Error Cases

- No saved reports found.
- Invalid Account.

Get Device Details in PDF

Users can download complete information of a particular device in pdf format. This includes patch details,compliance deviations etc,. The request can include either hostname or tag or both along with accountid. When both hostname and tag are entered, Saner Server will validate whether that tag is mapped with entered host.

An optional key “modules” added which accepts comma separated values “VM,CM,PM,AM”. If not provided, all modules for which data is available and is provisioned for the device will be downloaded.

Method Name: **getDevicePdfReport**

Method Type: POST

Mandatory Parameters: accountid and hostname

Sample request	Sample response
Request to get single device report: { "request": { "method": "getDevicePdfReport", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "account1" },{ "key": "hostname", "value": "my-pc" },{ "key": "tagname", "value": "tag01" },{ "key": "tagvalue", "value": "111-222-333-444" }] }] } } } Request to get multiple devices (upto 5) report: { "request": { "method": "getdevicepdfreport", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "Test_Account" }, { "key": "apifilters", "filters": [{ "tags": { "in": { "owner": ["User_1"] }, "not in": { "owner": ["User_1"] } } }] }] }] }	Response will be available in zip file which includes pdf and signature files.

Sample request	Sample response
<pre> } }</pre>	
<pre>Request to get reports with specific modules (VM, CM, PM, or AE): { "request": { "method": "getDevicePdfReport", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "account1" },{ "key": "hostname", "value": "host1" },{ "key": "modules", "value": "VM,CM,PM,AE" }] }] } } }</pre>	

Possible Error Cases

- Invalid Account.
- Invalid Input.
- Account Expired.
- Incomplete input.
- Device not found or saner not enabled.
- Exceeded limit of 5 device reports that can be downloaded in one request.
- Host <hostname> not found or agent is not enabled.

Apply Custom Report

This method will allow users to apply saved custom report from one account to another account or all existing and future accounts. Target account names can be multiple comma separated names under organization present in the field "targetorganizations" or it can be just “all”. When “targetaccounts” is given value as “all” it will be applied to all existing accounts where that report name does not exist and also to all future accounts that are created under that organization.

Method Name: addReportProvision

Method Type: POST

Mandatory Parameters: accountid, reportname, targetorganization, and targetaccounts

Sample request	Sample response
<pre>{ "request": {</pre>	<pre>{ "response": {</pre>

```
    "method": "addReportProvision",
    "parameters": {
      "parameterset": [{
        "parameter": [{
          "key": "accountid",
          "value": "sourceaccount"
        },{
          "key": "reportname",
          "value": "customreport1"
        },{
          "key": "targetorganization",
          "value": "organization1"
        },{
          "key": "targetaccounts",
          "value": "targetaccount1,
targetaccount2 | all"
        }]
      }]
    }
  }
}
```

```
    "method": "addReportProvision",
    "results": {
      "result": [{
        "key": "",
        "status": "SUCCESS",
        "reason": "Report customreport1
successfully applied to: targetaccount1"
      }]
    }
  }
}
```

Possible Error Cases

- Invalid Account.
- Report not found.
- Failed due to invalid account name or ID.
- Not enough inputs found.
- Invalid organization found
- Target accounts must not be empty.

Delete Custom Report

This method is used to delete the custom report of respective account and it will not be applied to the future accounts if it is marked to apply.

Method Name: deleteReportProvision

Method Type: POST

Mandatory Parameters: accountid, reportname, and organization

Sample request	Sample response
Request to delete single customer report: { "request": { "method": "deleteReportProvision", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "account1" },{	{ "response": { "method": "deleteReportProvision", "results": { "result": [{ "key": "", "status": "SUCCESS", "reason":"customreport1 successfully deleted" }] }

Sample request	Sample response
<pre>"key": "reportname", "value": "customreport1" },{ "key": "organization", "value": "organization1" }] }] } } }</pre>	<pre>} }</pre>
<pre>Request to delete multiple custom reports from multiple accounts: { "request": { "method": "deleteReportProvision", "parameters": { "parameterset": [{ "parameter": [{ "key": "accountid", "value": "account1" },{ "key": "reportname", "value": "report1" },{ "key": "organization", "value": "organization1" }] },{ "parameter": [{ "key": "accountid", "value": "account2" },{ "key": "reportname", "value": "report1,report2" },{ "key": "organization", "value": "organization2" }] }] } } }</pre>	

4

Possible Error Cases

- Account expired.
- Report not found.
- Not enough inputs found.
- Invalid Input or Default Report found which is not allowed to delete.
- Account not found under organization.

AD Integration Management

Saner can replicate the Organization hierarchy of Active Directory and synchronize the organizations, groups, and devices. AD integration on Saner enables automatic synchronization of devices getting commissioned/moved/decommissioned on Active Directory, ensuring their hierarchy in Saner is up to date. This helps in seamless deployment of Saner Agents. You can perform operations such as Adding AD Configuration, Updating AD Configuration, Deleting AD Configuration, and Testing AD Configuration using APIs.

This section provides insights into the Saner AD Integration APIs and their usage.

Add AD Configuration

This method allows you to configure AD to an organization in Saner Server. AD configuration could be attained via server or Saner Agent AD integration.

Request must be URL encoded when field SSL is passed. Field 'usecredential' and 'secureldap' are optional, if value not provided, then default values – *true* for 'usercredential and *false* for secureldap will be used.

Also, 'usecredential', field is used during agent integration only.

Method Name: addADconfig

Method Type: POST

Mandatory Parameters: All fields are mandatory.

Sample request	Sample response
<pre>{ "request": { "method": "addADconfig", "parameters": { "parameterset": [{ "parameter": [{ "key": "organization", "value": "testorganization1" },{ "key": "devicename", "value": "testdevice1" },{ "key": "integrationtype", "value": "server/agent" },{ "key": "save", "value": "yes/no" },{ "key": "username", "value": "ADuser1" },{ "key": "password", "value": "XXXXXXX" },{ "key": "domainname", "value": "ADdomain1.com"</pre>	In case of success: <pre>{ "response": { "method": "addADconfig", "results": { "result": [{ "key": "testorganization1", "status": "success", "reason": "" }] } } }</pre>

Sample request	Sample response
<pre>},{ "key": "ssl", "value": "sslcontent1" },{ "key": "sslverify", "value": "yes/no" },{ "key": "usecredential", "value": "true/false" },{ "key": "secureldap", "value": " true/false " },{ "key": "mode", "value": "auto/manual" },{ "key": "schedule", "value": "daily/weekly/monthly" },{ "key": "starttime", "value": "10:00:AM" },{ "key": "monthoftheyear", "value": "5" },{ "key": "weekofthemonth", "value": "1" },{ "key": "dayoftheweek", "value": "1" }] }] } } }</pre>	

Possible Error Cases

- AD configuration not added.
- Failed due to invalid organization.
- Failed due to invalid mode.
- Failed due to invalid integration type.
- Failed due to invalid save option.
- Failed due to invalid username.
- Failed due to invalid password.
- Failed due to invalid domain name.
- Failed due to invalid SSL.
- Failed due to invalid SSL verify option.
- Failed due to invalid start time.
- Failed due to invalid schedule.
- Failed due to invalid month of the year.
- Failed due to invalid week of the month.
- Failed due to invalid day of the week.
- Failed due to invalid day of the month.
- Organization does not exist.
- AD configuration already exists.
- Failed due to invalid schedule values.
- Failed due to invalid device name.

- AD credentials like domain name or username or password cannot be empty.
- Field <key> cannot be empty.

Update AD Configuration

This method allows you to update configured AD details of an organization in Saner Server. It requires name of organization and all other details.

Method Name: updateADconfig

Method Type: POST

Mandatory Parameters: organization.And depedning on the key you are looking to update, provide the value for the key.

Sample request	Sample response
<pre>{ "request": { "method": "updateADconfig", "parameters": { "parameterset": [{ "parameter": [{ "key": "organization", "value": "testorganization1" },{ "key": "devicename", "value": "testdevice1" },{ "key": "integrationtype", "value": "server/agent" },{ "key": "save", "value": "yes/no" },{ "key": "username", "value": "ADuser1" },{ "key": "password", "value": "ADpassword1" },{ "key": "domainname", "value": "ADdomain1.com" },{ "key": "ssl", "value": "sslcontent1" },{ "key": "sslverify", "value": "yes/no" },{ "key": "usecredential", "value": "true/false" },{ "key": "secureldap", "value": " true/false " },{ "key": "mode", "value": "auto/manual" },{</pre>	<pre>In case of success: { "response": { "method": "updateADconfig", "results": { "result": [{ "key": "testorganization1", "status": "success", "reason": "" }] } } }</pre>

Sample request	Sample response
<pre>"key": "schedule", "value": "daily/weekly/monthly" },{ "key": "starttime", "value": "10:00:AM" },{ "key": "monthoftheyear", "value": "5" },{ "key": "weekofthemonth", "value": "1" },{ "key": "dayoftheweek", "value": "1" }] }] } } }</pre>	

Possible Error Cases

- AD configuration not updated.
- Failed due to invalid organization.
- Failed due to invalid mode.
- Failed due to invalid integration type.
- Failed due to invalid save option.
- Failed due to invalid username.
- Failed due to invalid password.
- Failed due to invalid domain name.
- Failed due to invalid ssl.
- Failed due to invalid ssl verify option.
- Failed due to invalid start time.
- Failed due to invalid schedule.
- Failed due to invalid month of the year.
- Failed due to invalid week of the month.
- Failed due to invalid day of the week.
- Failed due to invalid day of the month.
- Organization does not exist.
- AD configuration does not exist.
- Failed due to invalid schedule values.
- Failed due to invalid device name.
- AD credentials like domain name or username or password cannot be empty.
- Field <key> cannot be empty.

Delete AD Configuration

This method allows you to remove AD configuration of the organizations in Saner Server.

It requires name of the organization to delete its AD configuration and field device name is required if AD was configured via agent integration.

Method Name: deleteADconfig

Method Type: POST

Mandatory Parameters: organization and devicename

Sample request	Sample response
<pre>{ "request": { "method": "deleteADconfig", "parameters": { "parameterset": [{ "parameter": [{ "key": "organization", "value": "testorganization1" },{ "key": "devicename", "value": "testdevice1" }] }] } } }</pre>	<pre>In case of success: { "response": { "method": "deleteADconfig", "results": { "result": [{ "key": "testorganization1", "status": "success", "reason": "" }] } } }</pre>

Possible Error Cases

- AD configuration not removed.
- Failed due to invalid organization.
- Failed due to invalid device name.
- Field organization cannot be empty.
- Organization does not exist.
- Field <key> cannot be empty.

Get AD Configuration

This feature allows you to get AD configuration details of the organizations in Saner Server. It requires name of organizations to get their AD configuration details. If no name is provided, all organization AD configuration details are fetched.

Method Name: getADconfig

Method Type: POST

Mandatory Parameters: organization

Sample request	Sample response
<pre>{ "request": { "method": "getADconfig", "parameters": { "parameterset": [{ "parameter": [{ "key": "organization", "value": "testorganization1" }] }] } } }</pre>	<pre>In case of success: { "adconfigurations": [{ "adconfigurationinfo": { "organization": "testorganization1", "integrationtype": "server", "adcredentials": { "name": "ADuser1", "password": "ADpassword1", "domainname": "ADdomain1.com", "secureldap": "false", "ssl": "", "sslverify": "no" }, "adschedule": { "starttime": "10:00:AM", "month-of-year": "5", "day-of-month": "", "week-of-month": "1", "day-of-week": "1" } } }] }</pre>
<pre>In order to get all organization AD configuration details: { "request": { "method": "getADconfig" } }</pre>	

Possible Error Cases

- Failed due to invalid organization.
- Field <key> cannot be empty.

Initiate AD Scan

This method allows you to initiate an AD scan for the Active Directory configured organizations in Saner Server. It requires name of the organization.

Method Name: initiateADscan

Method Type: POST

Mandatory Parameters: organization

Sample request	Sample response
<pre>{ "request": { "method": "initiateADscan", "parameters": { "parameterset": [{ "parameter": [{ "key": "organization", "value": "testorganization1" }] }] } } }</pre>	<pre>In case of success: { "response": { "method": "initiateADscan", "results": { "result": [{ "key": " testorganization1", "status": "success", "reason": "" }] } } }</pre>

Possible Error Cases

- AD scan not initiated.
- Failed due to invalid organization.
- Field organization cannot be empty.
- Organization does not exist.
- Failed due to invalid device name.
- Field <key> cannot be empty.

Download AD Agent

This method allows you to download AD agent installer for accounts under the organizations in Saner Server.

Method Name: downloadADAgent

Method Type: POST

Mandatory Parameters: organization and account

Sample request	Sample response
To download AD agent for an account: { "request": { "method": "downloadADagent", "parameters": { "parameterset": [{ "parameter": [{ "key": "organization", "value": "testorganization1" },{ "key": "account", "value": "testaccount1" }] }] } } }	In case of success, you would get a compressed file (ZIP format) with AD agent installers.
To get AD agent installer for all the accounts under an organization: { "request": { "method": "downloadADagent", "parameters": { "parameterset": [{ "parameter": [{ "key": "organization", "value": " testorganization1" }] }] } } }	

Possible Error Cases

- Failed due to invalid organization.
- Failed due to invalid account name.
- Organization does not exist.
- Field <key> cannot be empty.
- Account <account_name> does not exist under organization <organization_name>.

Test AD Connection

This method allows you to issue an AD connection test in the organizations in Saner Server. It requires name of the organization, integration type, AD username, AD password, AD domain name. Field device name is required if AD was configured via agent integration. If an organization is already AD configured, then only name of the organization is required to issue an AD connection test.

Field '*usecredential'* and '*secureldap*' are optional and if its value is not provided, then default value true and false will be used. Also, '*usecredential*' field is used during agent integration only.

If AD is configured, then only organization field is mandatory to issue test connection.

Method Name: testADconnection

Method Type: POST

Mandatory Parameters: organization

Sample request	Sample response
To issue an AD connection test: { "request": { "method": "testADconnection", "parameters": { "parameterset": [{ "parameter": [{ "key": "organization", "value": "testorganization1" },{ "key": "devicename", "value": "testdevice1" },{ "key": "integrationtype", "value": "server/agent" },{ "key": "username", "value": "ADuser1" },{ "key": "password", "value": "ADpassword1" },{ "key": "domainname", "value": "ADdomain1.com" },{ "key": "ssl", "value": "sslcontent1" },{ "key": "sslverify",	In case of success: { "response": { "method": "testADconnection", "results": { "result": [{ "key": "testorganization1", "status": "success", "reason": "" }] } } }

Sample request	Sample response
<pre>"value": "yes/no" },{ "key": "usecredential", "value": "true/false" },{ "key": "secureldap", "value": " true/false " }] }] } } }</pre>	
To issue AD connection test for an AD configured organization: <pre>{ "request": { "method": "testADconnection", "parameters": { "parameterset": [{ "parameter": [{ "key": "organization", "value": "testorganization1" }] }] } } }</pre>	

Possible Error Cases

- AD connection test not issued.
- Failed due to invalid organization.
- Failed due to invalid device name.
- Failed due to invalid integration type.
- Failed due to invalid username.
- Failed due to invalid password.
- Failed due to invalid domain name.
- Failed due to invalid ssl.
- Failed due to invalid ssl verify option.
- Organization does not exist.
- AD credentials like username or password or domainname cannot be empty.
- Field <key> cannot be empty.

Get AD Connection Status

This method allows you to get AD connection test status issued for the organizations in Saner Server. It requires the name of the organization.

Method Name: getADconnectionStatus

Method Type: POST

Mandatory Parameters: organization

Sample request	Sample response
<pre>{ "request": { "method": "getADconnectionStatus", "parameters": { "parameterset": [{ "parameter": [{ "key": "organization", "value": "testorganization1" }] }] } } }</pre>	In case of success: <pre>{ "response": { "method": "getADconnectionStatus", "results": { "result": [{ "key": "testorganization1", "status": "Success!", "reason": "" }] } } }</pre>

Possible Error Cases

- Failed due to invalid organization.
- Organization does not exist.
- No test connection status found.
- No status found.
- Invalid input
- Unable to connect to LDAP server.

Add entry to AD Exclude List

This method allows you to add items(site/group/device) to be excluded from the organizations in Saner Server, for which any changes based on AD scan won't be applied. It requires the name of the organization, site(s) or group(s) or device(s) to be excluded. Field "site" is required if a group or a device has to be excluded.

Method Name: addADscanExcludeList

Method Type: POST

Mandatory Parameters: site, organization, excludegroup or excludedevice

Sample request	Sample response
<pre>{ "request": { "method": "addADscanExcludeList", "parameters": { "parameterset": [{ "parameter": [{ "key": "organization", "value": " testorganization1" },{ "key": "site", "value": "testsite2" },{ "key": "excludegroup", "value": "testgroup1" },{ "key": "excludedevice",</pre>	In case of success: <pre>{ "response": { "method": "addADscanExcludeList", "results": { "result": [{ "key": "testorganization1", "status": "success", "reason": "" }] } } }</pre>

Sample request	Sample response
<pre>"value": "testdevice1" },{ "key": "excludegroup", "value": "testgroup2" },{ "key": "excludedevice", "value": "testdevice2" },{ "key": "excludesite", "value": "testsite1" },{ "key": "excludesite", "value": "testsite2" }] }] } } }</pre>	

Possible Error Cases

- Failed due to invalid organization.
- Exclude item cannot be empty.
- Exclude item not updated.
- Failed due to invalid site field.
- Failed due to invalid exclude device.
- Failed due to invalid exclude group.
- Failed due to invalid exclude site.
- Organization does not exist.
- AD configuration does not exist.
- Field <key> cannot be empty.

Remove entry from AD Exclude List

This method allows you to remove items(site/group/device) from exclude list of organizations in Saner Server.

Method Name: removeADscanExcludeList

Method Type: POST

Mandatory Parameters: organization, site (this key is mandatory is you want to remove a group or a device).

Sample request	Sample response
<pre>{ "request": { "method": "removeADscanExcludeList", "parameters": { "parameterset": [{ "parameter": [{ "key": "organization", "value": "testorganization1" },{</pre>	<pre>In case of success: { "response": { "method": "removeADscanExcludeList", "results": { "result": [{ "key": "testorganization1", "status": "success", "reason": ""</pre>

Sample request	Sample response
<pre>"key": "site", "value": "testsite2" },{ "key": "excludegroup", "value": "testgroup1" },{ "key": "excludedevice", "value": "testdevice1" },{ "key": "excludegroup", "value": "testgroup2" },{ "key": "excludedevice", "value": "testdevice2" },{ "key": "excludesite", "value": "testsite1" },{ "key": "excludesite", "value": "testsite2" }] }] } } }	<pre>}] } } }

Possible Error Cases

- No exclude items found.
- Exclude item cannot be empty.
- Failed due to invalid organization.
- Exclude item not updated.
- Failed due to invalid site field.
- Failed due to invalid exclude device.
- Failed due to invalid exclude group.
- Failed due to invalid exclude site.
- Organization does not exist.
- AD configuration does not exist.
- Field <key> cannot be empty.

Get all entries from AD Scan Exclude List

This method allows you to get AD scan excluded list of items for the organizations in Saner Server. It requires the name of the organization.

Method Name: getADscanExcludeList

Method Type: POST

Mandatory Parameters: organization

Sample request	Sample response
<pre>{ "request": {</pre>	<pre>In case of success: {</pre>

Sample request	Sample response
<pre>"method": "getADscanExcludeList", "parameters": { "parameterset": [{ "parameter": [{ "key": "organization", "value": " testorganization1" }] }] } } }</pre>	<pre>"ExcludedItems": [{ "name": "testsite1", "type": "site", "site": "" },{ "name": "testsite2", "type": "site", "site": "" },{ "name": "testgroup1", "type": "group", "site": "testsite2" },{ "name": "testgroup2", "type": "group", "site": "testsite2" },{ "name": "testdevice1", "type": "device", "site": "testsite2" },{ "name": "testdevice2", "type": "device", "site": "testsite2" }] }</pre>

Possible Error Cases

- Failed due to invalid organization.
- Field <key> cannot be empty.

Get AD Scan Merged Data

This method allows you to get AD scan merged data, which is the combination of existing data (sites, devices and groups) and AD scan data for the organizations in Saner Server. Organization name is a mandatory input for this method.

Method Name: getADscanMergedData

Method Type: POST

Mandatory Parameters: organization

Sample request	Sample response
<pre>{ "request": { "method": " getADscanMergedData", "parameters": { "parameterset": [{ "parameter": [{ "key": "organization", "value": "testorganization1"</pre>	<pre>In case of success: { "node": [{ "node": [{ "node": [{ "os": "Windows Server 2019 Standard 10.0 (17763)", "node": [], "ip": "192.168.xxx.xxx", "name": "testdevice1.domain",</pre>

Sample request	Sample response
}] }] } } }	"id": "data / ous / ou_name / testaccount1 / ou_gname / testgroup1 / ou_computers / ou_cname / testdevice1.domain", "type": "device", "status": "added" }], "name": "testgroup1", "id": "data / ous / ou_name / testaccount1 / ou_computers / ou_groups / ou_gname / testgroup1", "type": "group", "status": "added" }], "name": "testaccount1", "id": "data / ous / ou_name / testaccount1", "type": "ou", "status": "added" }], "name": "testorganization1", "id": "data / fqdn / testorganization1", "type": "fqdn", "status": "modified" }

Possible Error Cases

- Failed due to invalid organization.
- Field <key> cannot be empty.

Apply AD Scan Changes

This method allows you to apply AD scan changes manually for the organizations that exist in Saner Server. It requires the name of the organization and data, which is the AD scan merged data json. Remove the sites/groups/devices json section if changes related to it should not be processed; do not alter the json in any other way.

Method Name: applyADscanChanges

Method Type: POST

Mandatory Parameters: organization and data

Sample request	Sample response
To apply AD scan changes: { "request": { "method": "applyADscanChanges", "parameters": { "parameterset": [{ "parameter": [{ "key": "organization", "value": "testorganization1" },{ "key": "data", "value": "<ADscanMergedDataJson>" }]	In case of success: { "response": { "method": " applyADscanChanges ", "results": { "result": [{ "key": "testorganization1", "status": "success", "reason": "" }] } }

<pre>}] } } }</pre>	<pre>}</pre>

Json after removing the device json section from the <ADscanMergedDataJson>:


```
{
  "node": [{
    "node": [{
      "node": [],
      "name": "testgroup1",
      "id": "data / ous / ou_name / testaccount1 / ou_computers / ou_groups /
ou_gname / testgroup1",
      "type": "group",
      "status": "added"
    }],
    "name": "testaccount1",
    "id": "data / ous / ou_name / testaccount1",
    "type": "ou",
    "status": "added"
  }],
  "name": "testorganization1",
  "id": "data / fqdn / testorganization1",
  "type": "fqdn",
  "status": "modified"
}
```


Possible Error Cases

- AD scan changes not applied.
- Field organization and data cannot be empty.
- Failed due to invalid data.
- Failed due to invalid organization.
- Organization does not exist.
- Field <key> cannot be empty.

Get AD Scan Action Status

This method allows you to get AD scan action status for the organizations in Saner Server. Organization name is a mandatory input for this method.

Method Name: getADscanActionStatus

Method Type: POST

Mandatory Parameters: organization

Sample request	Sample response
<pre>{ "request": { "method": "getADscanActionStatus", "parameters": { "parameterset": [{</pre>	<pre>In case of success: { "response": { "method": "getADscanActionStatus",</pre>

<table>
<tr>
<td>

```
        "parameter": [{
          "key": "organization",
          "value": "testorganization1"
        }]
      }]
    }
  }
}
```

</td>
<td>

```
    "results": {
      "result": [{
        "key": "testorganization1",
        "status": "Success!",
        "reason": ""
      }]
    }
  }
}
```

</td>
</tr>
</table>

Possible Error Cases

- No AD scan action status found.
- Failed due to invalid organization.
- Organization does not exist.
- Field <key> cannot be empty.
- Invalid input

Network Scanner Integration Management

Saner Network Scanner identifies vulnerabilities across all IP-enabled devices within an Organization. You can perform automated periodic scans using Network Scanner and perform authenticated network scans. This section provides insights into the Saner Network Scanner APIs and their usage.

Add Network Scanner

This method allows users to add the device as a network scanner to an account. Account and device name are mandatory inputs for this method.

Method Name: addNetworkScanner

Method Type: POST

Mandatory Parameters: account and devicename

Sample request	Sample response
<pre>{ "request": { "method": "addNetworkScanner", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": "devicename", "value": "testdevice1" }] }] } } }</pre>	<pre>In case of success: { "response": { "method": "addNetworkScanner", "results": { "result": [{ "key": "testdevice1", "status": "success", "reason": "" }] } } }</pre>

Possible Error Cases

- Missing required parameters. Account and Device name must be provided.
- Field <key> cannot be empty.
- Failed due to invalid account name or Id.
- Failed due to invalid device name.
- Field account cannot be empty.
- Field device name cannot be empty.
- <devicename> is already a network scanner.
- Already issued an add network scanner request for <devicename>.
- <devicename> cannot be upgraded as a network scanner.

Remove Network Scanner

This method allows users to remove a network scanner role of a device. Account and device name as mandatory inputs for this method.

Method Name: removeNetworkScanner

Method Type: POST

Mandatory Parameters: account and devicename

Sample request	Sample response
<pre>{ "request": { "method": "removeNetworkScanner", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1 " }, { "key": "devicename", "value": "testdevice1" }] }] } } }</pre>	<pre>In case of success: { "response": { "method": "removeNetworkScanner", "results": { "result": [{ "key": "testdevice1", "status": "success", "reason": "" }] } } }</pre>

Possible Error Cases

- Missing required parameters. Account and Device name must be provided.
- Field <key> cannot be empty.
- Failed due to invalid account name or Id.
- Failed due to invalid device name.
- Field account cannot be empty.
- Field device name cannot be empty.
- <devicename> is not a network scanner.

Get status of network scan

This method allows users to fetch the status of a network scan. Account and device name as mandatory inputs for this method.

Method Name: getNetworkScanStatus

Method Type: POST

Mandatory Parameters: account and devicename

Sample request	Sample response
<pre>{ "request": { "method": "getNetworkScanStatus", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": "devicename", "value": "testdevice1" }] }] } } }</pre>	<pre>In case of success: { "response": { "method": "getNetworkScanStatus", "results": { "result": [{ "key": "testdevice1", "status": "network scan initiated", "reason": "" }] } } }</pre>

Possible Error Cases

- Invalid Input
- Missing required parameters. Account and Device name must be provided.
- Field <key> cannot be empty.
- Failed due to invalid account name or Id.
- Failed due to invalid device name.
- Field account cannot be empty.
- Field device name cannot be empty.
- No network scan status found.

Get status of discovery scan

This method allows users to fetch the status of a discovery scan. Account and device name as mandatory inputs for this method.

Method Name: **getDiscoveryScanStatus**

Method Type: POST

Mandatory Parameters: account and devicename

Sample request	Sample response
<pre>{ "request": { "method": "getDiscoveryScanStatus", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": "devicename", "value": "testdevice1" }]</pre>	<pre>In case of success: { "response": { "method": "getDiscoveryScanStatus", "results": { "result": [{ "key": "testdevice1", "status": "discovery scan initiated", "reason": "" }] }</pre>

<pre> }] } } }</pre>	<pre> } }</pre>

Possible Error Cases

- Invalid Input
- Missing required parameters. Account and Device name must be provided.
- Field <key> cannot be empty.
- Failed due to invalid account name or Id.
- Failed due to invalid device name.
- Field account cannot be empty.
- Field device name cannot be empty.
- No discovery scan status found.

Initiate Network Scan

This method allows users to initiate the network scan of a device. Account and device name as mandatory inputs for this method. To issue a network scan, we must create a network config and assign it to the network scanner device.

Method Name: initiateNetworkScan

Method Type: POST

Mandatory Parameters: account and devicename

Sample request	Sample response
<pre>{ "request": { "method": "initiateNetworkScan", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": "devicename", "value": "testdevice1" }] }] } } }</pre>	<pre>In case of success: { "response": { "method": "initiateNetworkScan", "results": { "result": [{ "key": "testdevice1", "status": "success", "reason": "" }] } } }</pre>

Possible Error Cases

- Missing required parameters. Account and Device name must be provided.
- Field <key> cannot be empty.
- Failed due to invalid account name or Id.
- Failed due to invalid device name.
- Field account cannot be empty.
- Field device name cannot be empty.
- <devicename> is not a network scanner.

Initiate Discovery Scan

This method allows users to initiate a discovery scan for a device. Account and device name as mandatory inputs for this method. To initiate a discovery scan, we must create a discovery config for the network scanner device. The "targets" field value must be passed for initiating an immediate discovery scan.

Method Name: initiateDiscoveryScan

Method Type: POST

Mandatory Parameters: account, targets, and devicename

Sample request	Sample response
<pre>{ "request": { "method": "initiateDiscoveryScan", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" }, { "key": "targets", "values": ["192.168.2.18"] }, { "key": "devicename", "value": "testdevice1" }] }] } } }</pre>	<pre>In case of success: { "response": { "method": "initiateDiscoveryScan", "results": { "result": [{ "key": "testdevice1", "status": "success", "reason": "" }] } } }</pre>

Possible Error Cases

- Missing required parameters. Account and Device name must be provided.
- Field <key> cannot be empty.
- Failed to initiate discovery scan.
- Field targets cannot be empty.
- Already a discovery scan is initiated.
- Already a discovery scan is ongoing.
- Failed due to invalid targets.
- Failed due to invalid account name or Id.
- Failed due to invalid device name.
- Field account cannot be empty.
- Field device name cannot be empty.
- <devicename> is not a network scanner.

Add Discovery Scan Config

This method allows users to add a discovery config for a network scanner. Account, device name, targets, and schedule field values as mandatory input fields for this method. Supported schedule field values are “daily”,” weekly”,” monthly”, and “date”.

Method Name: addDiscoveryScanConfig

Method Type: POST

Mandatory Parameters: account, devicename, targets, and schedule. Key’ schedule’ supported values – ‘daily’, ‘weekly, and ‘monthly’.

Sample request	Sample response
<pre>{ "request": { "method": "addDiscoveryScanConfig", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": "devicename", "value": "testdevice1" },{ "key": "targets", "values": ["192.168.XX.XX"] },{ "key": "target", "value": "www.testdomain1.com" },{ "key": "schedule", "value": "monthly" },{ "key": "starttime", "value": "08:10:AM" },{ "key": "endtime", "value": "09:10:AM" },{ "key": "monthoftheyear", "value": "5" },{ "key": "weekofthemonth", "value": "1" },{ "key": "dayoftheweek", "value": "1" }] }]}}}</pre>	<pre>In case of success: { "response": { "method": "addDiscoveryScanConfig", "results": { "result": [{ "key": "testdevice1", "status": "success", "reason": "" }] } } }</pre>

Possible Error Cases

- Missing required parameters. Account, Device name, Targets and Schedule Values must be provided.
- Field <key> cannot be empty.
- Failed due to invalid account name or Id.

- Failed due to invalid device name.
- Field account cannot be empty.
- Field device name cannot be empty.
- Failed due to invalid targets.
- Failed due to invalid start time.
- Failed due to invalid end time.
- Failed due to invalid schedule.
- Failed due to invalid month of the year.
- Failed due to invalid week of the month.
- Failed due to invalid day of the week.
- Failed due to invalid day of the month.
- Failed due to invalid date.
- Failed due to invalid schedule values.
- <devicename> is not a network scanner.
- Discovery config already exist for <devicename>.

Remove Discovery Scan Config

This method allows users to remove the discovery config of a network scanner device. Account name and device name as mandatory inputs for this method.

Method Name: removeDiscoveryScanConfig

Method Type: POST

Mandatory Parameters: account and devicename

Sample request	Sample response
<pre>{ "request": { "method": "removeDiscoveryScanConfig", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": "devicename", "value": "testdevice1" }] }] } } }</pre>	<pre>In case of success: { "response": { "method": "removeDiscoveryScanConfig", "results": { "result": [{ "key": "testdevice1", "status": "success", "reason": "" }] } } }</pre>

Possible Error Cases

- Missing required parameters. Account and Device name must be provided.
- Field <key> cannot be empty.
- Failed due to invalid account name or Id.
- Failed due to invalid device name.

- Field account cannot be empty.
- Field device name cannot be empty.
- Discovery config does not exist for <devicename>.

Remove Network Scan Config

This method allows users to remove a network scanner config. Account name and config name are mandatory inputs for this method.

Method Name: removeNetworkScanConfig

Method Type: POST

Mandatory Parameters: account and configname

Sample request	Sample response
<pre>{ "request": { "method": "removeNetworkScanConfig", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": "configname", "value": "testconfig1" }] }] } }</pre>	<pre>In case of success: { "response": { "method": "removeNetworkScanConfig", "results": { "result": [{ "key": "testconfig1", "status": "success", "reason": "" }] } } }</pre>

Possible Error Cases

- Network config not removed.
- Missing required parameters. Account and Config name must be provided.
- Field <key> cannot be empty.
- Failed due to invalid account name or Id.
- Failed due to invalid config name.
- Field account cannot be empty.
- Field config name cannot be empty.

Add Network Scan Config

This method allows users to add a network scanner config. Supported schedule field values are “daily”,” weekly” and” monthly”.

Method Name: addNetworkScanConfig

Method Type: POST

Mandatory Parameters: account, configname, target, targets, tcpports, udports, and portoption

Sample request	Sample response
<pre>{ "request": { "method": "addNetworkScanConfig", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": "configname", "value": "testconfig1" },{ "key": "description", "value": "config description" },{ "key": "targets", "values": ["192.168.XX.XX-XX"] },{ "key": "target", "value": "www.testdomain1.com" },{ "key": "excludetargets", "values": ["192.168.XX.XX"] },{ "key": "enableudpports", "value":"true" },{ "key": "tcpports", "values": ["80"] },{ "key": "udpports", "values": ["161"] },{ "key": "portoption", "value": "default ports" },{ "key": "schedule", "value": "monthly" },{ "key": "starttime", "value": "08:10:AM" },{ "key": "endtime", "value": "09:10:AM" },{ "key": "monthoftheyear", "value": "5" },{ "key": "weekofthemonth", "value": "1" },{ "key": "dayoftheweek", "value": "1" }] }]	<pre>In case of success: { "response": { "method": "addNetworkScanConfig "results": { "result": [{ "key": "testconfig1", "status": "success", "reason": "" }] } } }

Sample request	Sample response
} } }	

Possible Error Cases

- Network scan config not added.
- Missing required parameters. Account, Config name and Port must be provided.
- Name must be of minimum 4 characters.
- Name must be of maximum 50 characters.
- Field <key> cannot be empty.
- Failed due to invalid account name or Id.
- Failed due to invalid config name.
- Field account cannot be empty.
- Field config name cannot be empty.
- Failed due to invalid targets.
- Failed due to invalid exclude targets.
- Failed due to invalid description.
- Failed due to invalid port.
- Failed due to invalid start time.
- Failed due to invalid end time.
- Failed due to invalid schedule.
- Failed due to invalid month of the year.
- Failed due to invalid week of the month.
- Failed due to invalid day of the week.
- Failed due to invalid day of the month.
- Failed due to invalid schedule values.
- Failed due to invalid port option.
- Failed due to invalid udp ports.
- Failed due to invalid tcp ports.
- Port is mandatory. Either provide custom ports or available ports.
- Failed due to invalid tcp or udp ports.
- Failed due to invalid enable udp ports option. Value should be TRUE or FALSE.

Is Network Scanner

This method allows users to check if a device is a network scanner or not. It requires an account and device name as inputs.

Method Name: isNetworkScanner

Method Type: POST

Mandatory Parameters: account and device name

Sample request	Sample response
{ "request": { "method": "isNetworkScanner", "parameters": {	In case of success: { "response": { "method": "isNetworkScanner",

Sample request	Sample response
<pre>"parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": "devicename", "value": "testdevice1" }] }] } } }</pre>	<pre>"results": { "result": [{ "key": "testdevice1", "status": "True", "reason": "" }] } } }</pre>

Possible Error Cases

- Invalid Input
- Missing required parameters. Account and Device name must be provided.
- Field <key> cannot be empty.
- Failed due to invalid account name or Id.
- Failed due to invalid device name.
- Field account cannot be empty.
- Field device name cannot be empty.

Is Network Scan Config Assigned

This method allows users to check if a network scan config is assigned or not. Account, config name, and device name are mandatory inputs for this method.

Method Name: isNetworkScanConfigAssigned

Method Type: POST

Mandatory Parameters: account, configname, and devicename

Sample request	Sample response
<pre>{ "request": { "method": "isNetworkScanConfigAssigned", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": "configname", "value": "testconfig1" },{ "key": "devicename", "value": "testdevice1" }] }] }</pre>	<pre>In case of success: { "response": { "method": "isNetworkScanConfigAssigned", "results": { "result": [{ "key": "testdevice1", "status": "True", "reason": "" }] } } }</pre>

Sample request	Sample response
} }	

Possible Error Cases

- Invalid Input.
- Missing required parameters. Account, Config and Device name must be provided.
- Field <key> cannot be empty.
- Failed due to invalid account name or Id
- Failed due to invalid device name.
- Field account cannot be empty.
- Field device name cannot be empty.
- Failed due to invalid config name.
- Field config name cannot be empty.

Assign Network Scan Config

This method allows users to assign a network scan config to a device. Account name, config name, and device name are mandatory inputs for this method.

Method Name: assignNetworkScanConfig

Method Type: POST

Mandatory Parameters: account, configname, and devicename

Sample request	Sample response
<pre>{ "request": { "method": "assignNetworkScanConfig", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": "devicename", "value": "testdevice1" },{ "key": "configname", "value": "testconfig1" }] }] } } }</pre>	<pre>In case of success: { "response": { "method": " assignNetworkScanConfig ", "results": { "result": [{ "key": "testdevice1", "status": "success", "reason": "" }] } } }</pre>

Possible Error Cases

- Field <key> cannot be empty.
- Missing required parameters. Account, Config and Device name must be provided.
- Failed due to invalid account name or Id.
- Failed due to invalid config name.
- Failed due to invalid device name.

- Field account cannot be empty.
- Field device name cannot be empty.
- Field config name cannot be empty.
- <devicename> is not a network scanner.
- <devicename> already has a network scan config.

Unassign Network Scan Config

This method allows users to remove a network scanner config from a device. Account name and device name are mandatory inputs for this method.

Method Name: unassignNetworkScanConfig

Method Type: POST

Mandatory Parameters: account and devicename

Sample request	Sample response
<pre>{ "request": { "method": "unassignNetworkScanConfig", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": "devicename", "value": "testdevice1" }] }] } }}</pre>	<pre>In case of success: { "response": { "method": " unassignNetworkScanConfig ", "results": { "result": [{ "key": "testdevice1", "status": "success", "reason": "" }] } } }</pre>

Possible Error Cases

- Field <key> cannot be empty.
- Missing required parameters. Account and Device name must be provided.
- Failed due to invalid account name or Id
- Failed due to invalid config name.
- Failed due to invalid device name.
- Field account cannot be empty.
- Field device name cannot be empty.
- Field config name cannot be empty.
- <devicename> has no network scan config assigned.
- Cannot unassign network scan config. Already a network scan is ongoing.
- Cannot unassign network scan config. Already a network scan is initiated.
- Cannot unassign network scan config. Already a network scan is issued.

Get Network Scan Policy

This method allows users to get details of a network scan policy. Account name and policy name are mandatory inputs for this method.

Method Name: getNetworkScanPolicy

Method Type: POST

Mandatory Parameters: account and policyname

Sample request	Sample response
{ "request": { "method": "getNetworkScanPolicy", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": "policyname", "value": "Policy2" }] }] } } }	In case of success: { "networkPolicy": { "credentials": [{ "auth_type": "", "inputs": [{ "name": "WebApp Path", "sensitivity": "false", "value": "/myblog" }], "type": "script_preference" }], "name": "Policy2", "description": "desc", "scripts": { "Web Application": [{ "affectedVersion": "", "references": [], "scores": {}, "description": "Check the presence of MySQL Enterprise Monitor. This script sends an HTTP GET request and attempts to check the presence of MySQL Enterprise Monitor, tries to fetch, and register its version.", "productName": "MySQL Enterprise Monitor", "reliable": "Reliable", "solution": "", "name": "MySQL Enterprise Monitor detection", "id": "100390", "family": "Web Application", "category": ["default", "discovery", "safe", "version"], "cveReferences": ["CVE-2015-3144", "CVE-2016-5590", "CVE-2018-11040", "CVE-2017-5645", "CVE-2020-11996", "CVE-2017-3307", "CVE-2020-1935", "CVE-2017-3306", "CVE-2019-1551", "CVE-2016-3461", "CVE-2013-4316", "CVE-2020-1967", "CVE-2018-1258", "CVE-2017-10424", "CVE-2020-9484", "CVE-2021-23841"], "riskFactor": "" }], "Application Server": [{ "affectedVersion": "", "references": [], "scores": {}, "description": "Check the presence of WildFly Application Server. This script sends an HTTP GET request and attempts to check the presence of WildFly Application Server, tries to fetch and register its version.",

Sample request	Sample response
	"productName": "WildFly Application Server", "reliable": "Reliable", "solution": "", "name": "WildFly Application Server detection", "id": "100186", "family": "Application Server", "category": ["default", "discovery", "safe", "version"], "cveReferences": ["CVE-2020-27822", "CVE-2020-25640", "CVE-2020-10718", "CVE-2020-10740", "CVE-2020-25689", "CVE-2019-3805", "CVE-2018-14627", "CVE-2021-3536", "CVE-2020-1719", "CVE-2019-14887", "CVE-2019-3894"], "riskFactor": "" }] } } }

Possible Error Cases

- No Records.
- Missing required parameters. Account and Policy name must be provided.

Remove Network Scan Policy

This method allows users to remove the network scan policy. Account name and policy name are mandatory inputs for this method.

Method Name: removeNetworkScanPolicy

Method Type: POST

Mandatory Parameters: account and policyname

Sample request	Sample response
{ "request": { "method": "removeNetworkScanPolicy", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": "policyname", "value": "Policy3" }] }] } } }	{ "response": { "method": "removeNetworkScanPolicy", "results": { "result": [{ "key": "Policy3", "status": "success", "reason": "" }] } } }

Possible Error Cases

- Network scan not stopped.
- Missing required parameters. Account, Policy name must be provided.
- Default Policy cannot be removed.
- Field <key> cannot be empty.
- Failed due to invalid account name or Id.
- Failed due to invalid device name.
- Field account cannot be empty.
- Field device name cannot be empty.
- <devicename> is not a network scanner.
- Scan is not initiated or completed already.
- Scan is not ongoing.

Assign Network Scan Policy

This method allows users to assign a network scan policy to a device. Account name and policy name are mandatory inputs for this method.

Method Name: **assignNetworkScanPolicy**

Method Type: POST

Mandatory Parameters: account, devicename, and policyname

Sample request	Sample response
<pre>{ "request": { "method": "assignNetworkScanPolicy", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": "devicename", "value": "testdevice1" },{ "key": "policyname", "value": "Policy1" }] }] } } }</pre>	<pre>{ "response": { "method": "assignNetworkScanPolicy", "results": { "result": [{ "key": "testdevice1", "status": "success", "reason": "" }] } } }</pre>

Possible Error Cases

- Network scan not stopped.
- Missing required parameters. Account, Policy, and Device name must be provided.
- Invalid Input.
- Field <key> cannot be empty.
- Failed due to invalid account name or ID.
- Failed due to invalid device name.
- Field account cannot be empty.
- Field device name cannot be empty.

- Failed due to invalid policy name.
- Field policy name cannot be empty.
- <devicename> is not a network scanner.
- <policyname> is already assigned to <devicename>.
- Cannot assign network scan policy. Already a network scan is ongoing.
- Cannot assign network scan policy. Already a network scan is initiated.
- Cannot update network scan config. Already a network scan is issued.

Is Network Scan Policy Assigned

This method allows users to check if a scan policy is assigned or not. Account, policy, and device names are mandatory inputs for this method.

Method Name: **isNetworkScanPolicyAssigned**

Method Type: POST

Mandatory Parameters: account, policyname, and devicename

Sample request	Sample response
<pre>{ "request": { "method": "isNetworkScanPolicyAssigned", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": "policyname", "value": "Policy1" },{ "key": "devicename", "value": "testdevice1" }] }] } } }</pre>	<pre>{ "response": { "method": "isNetworkScanPolicyAssigned", "results": { "result": [{ "key": "testdevice1", "status": "True", "reason": "" }] } } }</pre>

Possible Error Cases

- Invalid Input.
- Field <key> cannot be empty.
- Missing required parameters. Account, Policy, and Device name must be provided.
- Failed due to invalid account name or Id.
- Failed due to invalid device name.
- Field account cannot be empty.
- Field device name cannot be empty.

Get All Network Scan Policy Names

This method allows users to get a list of all network scan policy names. Account Name is a mandatory input for this method.

Method Name: getAllNetworkScanPolicyNames

Method Type: POST

Mandatory Parameters: account

Sample request	Sample response
<pre>{ "request": { "method": "getAllNetworkScanPolicyNames", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" }] }] } } }</pre>	<pre>{ "networkPolicyNames": [{ "account": "testaccount1", "policyNames": ["Default Policy", "Policy1", "Policy2"] }] }</pre>

Possible Error Cases

- Field <key> cannot be empty.
- Missing required parameter. Account must be provided.
- Failed due to invalid account name or ID.

Get All Network Scan Config Names

This method allows users to get a list of all network scan config names. Account Name is a mandatory input for this method.

Method Name: getAllNetworkScanConfigNames

Method Type: POST

Mandatory Parameters: account

Sample request	Sample response
<pre>{ "request": { "method": "getAllNetworkScanConfigNames", "parameters": { "parameterset": [{</pre>	<pre>{ "networkScanConfigNames": [{ "account": "testaccount1", "configNames": ["Config1", "Config2"] }] }</pre>

Sample request	Sample response
<pre>"parameter": [{ "key": "account", "value": "testaccount1" }] }] } } }</pre>	

Possible Error Cases

- Invalid Input
- Missing required parameter. Account must be provided.
- Field <key> cannot be empty.
- Failed due to invalid account name or Id.
- Failed due to invalid device name.
- Field account cannot be empty.
- Field device name cannot be empty.
- No remove network scanner status found.

Stop Discovery Scan

This method allows users to stop the discovery scan. Account Name and device names must be provided as mandatory inputs for this method.

Method Name: stopDiscoveryScan

Method Type: POST

Mandatory Parameters: account and devicename

Sample request	Sample response
<pre>{ "request": { "method": "stopDiscoveryScan", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": "devicename", "value": "testdevice1" }] }] } } }</pre>	<pre>{ "response": { "method": "stopDiscoveryScan", "results": { "result": [{ "key": "testdevice1", "status": "success", "reason": "" }] } } }</pre>

Possible Error Cases

- Discovery scan not stopped.
- Missing required parameters. Account and Device name must be provided.
- Field <key> cannot be empty.

- Failed due to invalid account name or Id.
- Failed due to invalid device name.
- Field account cannot be empty.
- Field device name cannot be empty.
- <devicename> is not a network scanner.
- Scan is not initiated or completed already.
- Scan is not ongoing.

Stop Network Scan

This method allows users to stop the network scan. Account Name and device name are mandatory inputs for this method.

Method Name: stopNetworkScan

Method Type: POST

Mandatory Parameters: account and devicename

Sample request	Sample response
<pre>{ "request": { "method": "stopNetworkScan", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": "devicename", "value": "testdevice1" }] }] } } }</pre>	<pre>{ "response": { "method": "stopNetworkScan", "results": { "result": [{ "key": "testdevice1", "status": "success", "reason": "" }] } } }</pre>

Possible Error Cases

- Network scan not stopped.
- Missing required parameters. Account and Device name must be provided.
- Field <key> cannot be empty.
- Failed due to invalid account name or Id.
- Failed due to invalid device name.
- Field account cannot be empty.
- Field device name cannot be empty.
- <devicename> is not a network scanner.
- Scan is not initiated or completed already.
- Scan is not ongoing.

Get Network Scan Config

This method allows users to get network scan config. Account Name and config name are mandatory inputs for this method.

Method Name: getNetworkScanConfig

Method Type: POST

Mandatory Parameters: account and configname

Sample request	Sample response
<pre>{ "request": { "method": "getNetworkScanConfig", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": configname", "value": "Config1" }] }] } } }</pre>	<pre>{ "networkConfig": { "name": "Config1", "description": "desc", "portOption": "Top 100", "targets": ["192.168.2.XX-XX"], "excludeTargets": ["192.168.XX.XX"], "udpPorts": ["161"], "tcpPorts": ["80"], "schedule": { "type": "monthly", "date": "", "starttime": "11:00:AM", "endtime": "12:00:PM", "monthoftheyear": "2", "dayofthemonth": "", "weekofthemonth": "2", "dayoftheweek": "1" } } }</pre>

Possible Error Cases

- Field <key> cannot be empty.
- Missing required parameters. Account and Config name must be provided.
- Failed due to invalid account name or Id.
- Failed due to invalid config name.

Update Network Scan Config

This method allows users to update network scan config. Account and config names are mandatory inputs for this method.

Method Name: updateNetworkScanConfig

Method Type: POST

Mandatory Parameters: account, configname, and newconfigname

Sample request	Sample response
{ "request": { "method": "updateNetworkScanConfig", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" }, { "key": "configname", "value": "Config1" }, { "key": "newconfigname", "value": "NewConfig1" }, { "key": "enableudpports", "value":"true" }, { "key": "description", "value": "config description" }, { "key": "targets", "values": ["192.168.XX.XX-XX"] }, { "key": "excludetargets", "values": ["192.168.XX.XX"] }, { "key": "schedule", "value": "daily" }, { "key": "starttime", "value": "08:10:AM" }, { "key": "endtime", "value": "09:10:AM" }, { "key": "tcpports", "values": ["443"] }, { "key": "udpports", "values": ["533"	{ "response": { "method": " updateNetworkScanConfig ", "results": { "result": [{ "key": "NewConfig1", "status": "success", "reason": "" }] } } }

Sample request	Sample response
<pre>] }, { "key": "portoption", "value": "default ports" }] }] } } }</pre>	

Possible Error Cases

- Network scan config not updated.
- Missing required parameters. Account and Config name must be provided.
- Name must be of minimum 4 characters.
- Name must be of maximum 50 characters.
- Field <key> cannot be empty.
- Failed due to invalid account name or Id.
- Failed due to invalid config name.
- Field account cannot be empty.
- Field config name cannot be empty.
- Failed due to invalid targets.
- Failed due to invalid exclude targets.
- Failed due to invalid description.
- Failed due to invalid port.
- Failed due to invalid start time.
- Failed due to invalid end time.
- Failed due to invalid schedule.
- Failed due to invalid month of the year.
- Failed due to invalid week of the month.
- Failed due to invalid day of the week.
- Failed due to invalid day of the month.
- Failed due to invalid schedule values.
- Failed due to invalid new config name.
- Failed due to invalid remove exclude target option.
- Failed due to duplicate config name.
- <configname> does not exist.
- Cannot update network scan config. Already a network scan is ongoing.
- Cannot update network scan config. Already a network scan is initiated.
- Cannot update network scan config. Already a network scan is issued.
- Failed due to invalid enable udp ports option. Value should be TRUE or FALSE.

Get Discovery Scan Config

This method allows users to get the discovery config. Account and device names must be provided as mandatory inputs for this method.

Method Name: **getDiscoveryScanConfig**

Method Type: POST

Mandatory Parameters: account and devicename

Sample request	Sample response
<pre>{ "request": { "method": "getDiscoveryScanConfig", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1" },{ "key": "devicename", "value": "testdevice1" }] }] } } }</pre>	<pre>In case of success: { "discoveryConfig": { "targets": ["192.168.XX.XX"], "schedule": { "type": "daily", "date": "", "starttime": "08:10:AM", "endtime": "09:10:AM", "monthoftheyear": "1,2,3,4,5,6,7,8,9,10,11,12", "dayofthemonth": "1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21, 22,23,24,25,26,27,28,29,30,31", "weekofthemonth": "", "dayoftheweek": "" } } }</pre>

Possible Error Cases

- Field <key> cannot be empty.
- Missing required parameters. Account and Device name must be provided.
- Failed due to invalid account name or ID.
- Failed due to invalid device name.

Update Discovery Scan Config

This method allows users to update the discovery configuration. Supported schedule types for update discovery config is -daily/weekly/monthly. Account and device names must be provided as mandatory inputs for this method. Key ‘schedule’ accepts ‘daily’, ‘weekly’, and ‘monthly’ as values.

Method Name: updateDiscoveryScanConfig

Method Type: POST

Mandatory Parameters: account and devicename

Sample request	Sample response
<pre>{ "request": { "method": "updateDiscoveryScanConfig", "parameters": { "parameterset": [{ "parameter": [{ "key": "account", "value": "testaccount1"</pre>	<pre>{ "response": { "method": "updateDiscoveryScanConfig", "results": { "result": [{ "key": "testdevice1", "status": "success", "reason": ""</pre>

Sample request	Sample response
<pre>},{ "key": "devicename", "value": "testdevice1" },{ "key": "targets", "values": ["192.168.XX.XX"] },{ "key": "schedule", "value": "monthly" },{ "key": "starttime", "value": "02:10:AM" },{ "key": "endtime", "value": "05:10:AM" },{ "key": "monthoftheyear", "value": "5" },{ "key": "weekofthemonth", "value": "1,2" },{ "key": "dayoftheweek", "value": "1" }] }] } }}	<pre>}] } } }

Possible Error Cases

- Missing required parameters. Account and Device name must be provided.
- Field <key> cannot be empty.
- Failed due to invalid account name or ID.
- Failed due to invalid device name.
- Field account cannot be empty.
- Field device name cannot be empty.
- Failed due to invalid targets.
- Failed due to invalid start time.
- Failed due to invalid end time.
- Failed due to invalid schedule.
- Failed due to invalid month of the year.
- Failed due to invalid week of the month.
- Failed due to invalid day of the week.
- Failed due to invalid day of the month.
- Failed due to invalid date.
- Failed due to invalid schedule values.
- <devicename> is not a network scanner.
- Discovery config does not exist for <devicename>.

Multi-Factor Authentication

Saner Server can be secured by using an additional layer of security by enabling Multi-factor Authentication. Saner supports single sign-on vendors like PingOne, PingID, and Okta and time-based one-time password (TOTP) authentication applications like Google Authenticator, Microsoft Authenticator, etc.This section provides insights into Saner multi-factor APIs and their usage.

Add MFA Policy

This method allows users to add a multi-factor authentication policy. To add a multi-factor authentication policy, user needs policy name, policy description, environment ID, client ID, authentication path, and Ping One username.

Method Name: addMFAPolicy

Method Type: POST

Mandatory Parameters: policyName, policyDescription, environmentID, ClientID, authenticationPath, and usernameOption

Sample request	Sample response
<pre>{ "request": { "method": "addMFAPolicy", "parameters": { "parameterset": [{ "parameter": [{ "key": "policyName", "value": "testPolicy1" }, { "key": "policyDescription", "value": "testPolicy1 desc" }, { "key": "environmentId", "value": "xxxxxx-xxxx-xxxx-xxxx- xxxxxxxxxxx" }, { "key": "clientId", "value": " xxxxxx-xxxx-xxxx-xxxx- xxxxxxxxxxx " }, { "key": "authenticationPath", "value": "https://auth.pingone.asia" }, { "key": "usernameOption", "value": "loginid" }] }] } } }</pre>	<pre>{ "response": { "method": "addMFAPolicy", "results": { "result": [{ "key": "testPolicy1", "status": "success", "reason": "" }] } } }</pre>

Possible Error Cases

- Field <key> cannot be empty.
- Multi-Factor Authentication policy is not added.
- Invalid multi-factor authentication policy name.
- Invalid multi-factor authentication policy description.
- Invalid username option.
- Invalid multi-factor authentication input.
- Multi-Factor Authentication policy name already exist.

Update MFA Policy

This method allows users to update a multi-factor authentication policy. To update the multi-factor authentication policy user needs policy name, new policy name, policy description, environment ID, client ID, authentication path, PingOne username and new policy name field value is optional.

Method Name: updateMFAPolicy

Method Type: POST

Mandatory Parameters: policyName, newPolicyName, policyDescription, environmentId, clientId, authenticationPath, and usernameOption

Sample request	Sample response
{ "request": { "method": "updateMFAPolicy", "parameters": { "parameterset": [{ "parameter": [{ "key": "policyName", "value": "testPolicy1" },{ "key": "newPolicyName", "value": "newTestPolicy1" } ,{ "key": "policyDescription", "value": "testPolicy1 desc" }, { "key": "environmentId", "value": " xxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx " }, { "key": "clientId", "value": " xxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx " }, { "key": "authenticationPath", "value": "https://auth.pingone.asia" }, { "key": "usernameOption", "value": "loginid" }] }] } } }	{ "response": { "method": "updateMFAPolicy", "results": { "result": [{ "key": "newTestPolicy1", "status": "success", "reason": "" }] } } }

Possible Error Cases

- Field <key> cannot be empty.
- Multi-Factor Authentication policy is not updated.
- Multi-Factor Authentication default policy cannot be updated.
- Invalid multi-factor authentication policy name.
- Invalid multi-factor authentication policy description.
- Invalid username option.
- Invalid multi-factor authentication input.
- Multi-Factor Authentication policy does not exist.

Remove MFA Policy

This method allows users to remove the created multi-factor authentication policy. It requires only the policy name to delete the created multi-factor authentication policy.

Method Name: removeMFAPolicy

Method Type: POST

Mandatory Parameters: policyName

Sample request	Sample response
<pre>{ "request": { "method": "removeMFAPolicy", "parameters": { "parameterset": [{ "parameter": [{ "key": "policyName", "value": "testPolicy1" }] }] } } }</pre>	<pre>{ "response": { "method": "removeMFAPolicy", "results": { "result": [{ "key": "testPolicy1", "status": "success", "reason": "" }] } } }</pre>

Possible Error Cases

- Field <key> cannot be empty.
- Multi-Factor Authentication policy is not removed.
- Invalid multi-factor authentication policy name.
- Multi-Factor Authentication policy does not exist.

Get MFA Policy

This method allows users to get details of existing multi-factor authentication policies. It requires policy name to verify multi-factor policy existence.

Method Name: getMFAPolicy

Method Type: POST

Mandatory parameters: policyName

Sample request	Sample response
<pre>{ "request": { "method": "getMFAPolicy", "parameters": { "parameterset": [{ "parameter": [{ "key": "policyName", "value": "testPolicy1" }] }] } } }</pre>	<pre>{ "mfaPolicy": { "provider": "PingOne", "name": "testPolicy1", "description": "testPolicy1 desc", "config": { "environmentId": " xxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx ", "clientId": " xxxxxx-xxxx-xxxx-xxxx-xxxxxxxxxxxx ", "usernameOption": "loginid", "authenticationPath": "https://auth.pingone.asia" } } }</pre>

Possible Error Cases:

- No records found with given name.

Is MFA Policy Exists

This method allows users to identify if a multi-factor authentication policy exists or not. It only requires a policy name to get its details.

Method Name: **isMFAPolicyExist**

Method Type: POST

Mandatory parameters: policyName

Sample request	Sample response
<pre>{ "request": { "method": "isMFAPolicyExist", "parameters": { "parameterset": [{ "parameter": [{ "key": "policyName", "value": "testPolicy1" }] }] }}}</pre>	<pre>{ "response": { "method": "isMFAPolicyExist", "results": { "result": [{ "key": "testPolicy1", "status": "True", "reason": "" }] } } }</pre>

Possible Error Cases:

- Invalid Input.

Update User MFA Policy

This method allows users to update the assigned multi-factor authentication policy into a new one. It requires policy name of users and login id to update.

Method Name: updateUserMFAPolicy

Method Type: POST

Mandatory parameters: loginID, policyName

Sample request	Sample response
{ "request": { "method": "updateUserMFAPolicy", "parameters": { "parameterset": [{ "parameter": [{ "key": "policyName", "value": "testPolicy1" }, { "key": "loginId", "value": "testuser1@domain.com" }] }] } } }	{ "response": { "method": "updateUserMFAPolicy", "results": { "result": [{ "key": "testuser1@domain.com", "status": "success", "reason": "" }] } } }

Possible Error Cases

- Field <key> cannot be empty.
- Multi-Factor Authentication policy of user is not updated.
- Failed due to invalid login id.
- Multi-Factor Authentication policy does not exist.
- Multi-Factor Authentication policy does not exist for user.

Get User MFA Policy

This method allows users to get the name of the multi-factor authentication policy assigned to a user. It requires only login id of user to get the details.

Method Name: getUserMFAPolicy

Method Type: POST

Mandatory parameters: loginId

Sample request	Sample response

<table>
<tr>
<td>

```
{
  "request": {
    "method": "getUserMFAPolicy",
      "parameters": {
      "parameterset": [{
         "parameter": [{
           "key": "loginId",
             "value":
"testuser1@domain.com"
         }]
       }]
  }
  }
}
```

</td>
<td>

```
{
  "policyName": "testPolicy1"
}
```

</td>
</tr>
</table>

Possible Error Cases

- No records found with given name.

Is User MFA Policy Exists

This method allows users to identify if any multi-factor authentication policy is assigned to a user. It requires only login id to identify existence.

Method Name: isUserMFAPolicyExist

Method Type: POST

Mandatory Parameters: loginId

<table>
<tr>
<th>Sample request</th>
<th>Sample response</th>
</tr>
<tr>
<td>

```
{
  "request": {
    "method": "isUserMFAPolicyExist",
      "parameters": {
      "parameterset": [{
        "parameter": [{
        "key": "loginId",
        "value":
"testuser1@domain.com"
        }]
       }]
   }}
}
```

</td>
<td>

```
{
  "response": {
    "method": "isUserMFAPolicyExist",
      "results": {
      "result": [{
      "key": "testuser1@domain.com",
       "status": "True",
       "reason": ""
      }]
   }
  }
}
```

</td>
</tr>
</table>

Possible warning Cases

- Biometrics and Security key authentication are not supported. User must update supported authentication method in PingOne to proceed.

Enforce Multi Factor For User

This method is used to enforce multi-factor authentication for the user. If value is passed as "all" for login id then multi-factor authentication will be enforced for all the users under the ADMIN, including the admin,

otherwise if a specific user login id is passed then multi-factor will be enforced only to the specified user. It requires only the login id of the user. By default, policy name field will have "Google Authenticator" MFA policy as its value.

Method Name: enforceMultiFactor

Method Type: POST

Mandatory Parameters: loginid, policyName

Sample request	Sample response
<pre>{ "request": { "method": "enforceMultiFactor", "parameters": { "parameterset": [{ "parameter": [{ "key": "loginid", "value": "testuser12@secpod.com" }, { "key": "policyName", "value": "Authenticator App" }] }] } } }</pre>	<pre>{ "response": { "method": "enforceMultiFactor", "results": { "result": [{ "key": "testuser12@secpod.com", "status": "success", "reason": "" }] } } }</pre>

Possible Error Cases

- No user found to enforce multi-factor authentication.
- Invalid multi-factor authentication policy name

Withdraw Multi Factor For User

This method is used to withdraw multi-factor authentication for the user. If value is passed as "all" for login id then multi-factor authentication will be withdrawn for all the users under the ADMIN, including the admin, otherwise if a specific user login id is passed then multi-factor will be withdrawn only to the specified user. It requires only the login id of the user.

Method Name: withdrawMultiFactor

Method Type: POST

Mandatory Parameters: loginid

Sample request	Sample response

<table>
<tr>
<td>

```
{
  "request": {
    "method": "withdrawMultiFactor",
    "parameters": {
      "parameterset": [{
      "parameter": [{
        "key": "loginid",
        "value":
"testloginid@domain.com"
        }]
      }]
 }
  }
}
```

</td>
<td>

```
{
  "response": {
    "method": "withdrawMultiFactor",
    "results": {
      "result": [{
      "key": "testloginid@domain.com",
"status": "success",
      "reason": ""
      }]
 }
  }
}
```

</td>
</tr>
</table>

Possible Error Cases:

- Withdrawing multi-factor authentication not updated.
- Filed <key> cannot be empty.
- Failed due to invalid login id.
- Failed loginid must be provided.

Is Multi Factor Enforced For User

This method is used to verify multi-factor authentication is enforced for the user. It requires only the login id of the user.

Method Name: isMultiFactorEnforced

Method Type: POST

Mandatory Parameters: loginid

<table>
<tr><th>Sample request</th><th>Sample response</th></tr>
<tr>
<td>

```
{
  "request": {
    "method": "isMultiFactorEnforced",
    "parameters": {
      "parameterset": [{
      "parameter": [{
          "key": "loginid",
          "value":
"testloginid@domain.com"
        }]
       }]
 }
  }
}
```

</td>
<td>

```
{
  "response": {
    "method": "isMultiFactorEnforced",
    "results": {
      "result": [{
        "key":
"testloginid@domain.com",
        "status": "True",
"reason": ""
      }]
  }
  }
}
```

</td>
</tr>
</table>

Possible Error Cases:

- Invalid Input
- Filed <key> cannot be empty.
- Failed due to invalid login id.
- Field logninId cannot be empty.
- No status found.

Is Multi Factor Enabled For User

This method is used to check multi-factor authentication is enabled for the user. It requires only the login id of the user.

Method Name: isMultiFactorEnabled

Method Type: POST

Mandatory Parameters: loginid

Sample request	Sample response
<pre>{ "request": { "method": "isMultiFactorEnabled", "parameters": { "parameterset": [{ "parameter": [{ "key": "loginid", "value": "testloginid@domain.com" }] }] } } }</pre>	<pre>{ "response": { "method": "isMultiFactorEnabled", "results": { "result": [{ "key": "testloginid@domain.com", "status": "True", "reason": "" }] } } }</pre>

Possible Error Cases:

- Invalid Input
- Filed <key> cannot be empty.
- Failed due to invalid login id.
- Field logninId cannot be empty.
- Access denied.
- No status found.

Changelog

Release 6.6

Newly Added API

New API	Description
pinandunpindevice	This API allows administrators to pin or unpin a device from a group.
movedevice	This API allows administrators to move one or more devices across organizations, accounts, or groups, with an option to pin the devices to the

	target group to retain group association regardless of grouping criteria.
excludevulnerability	This API allows administrators to exclude vulnerabilities (CVEs), patches, or assets from remediation and visibility based on the defined scope, with exclusions applied at the account, group, or device level for a specified duration and automatically removed upon expiry.
addBlackListedAssets	This API allows administrators to mark one or more applications as blacklisted. Blacklisted applications detected on endpoints are flagged in the Asset Exposure (AE) dashboard, enabling administrators to identify violations and take appropriate action.
addWhiteListedAssets	This API allows administrators to mark one or more applications to the whitelist, marking them as approved and trusted while overriding any existing blacklist entries in the Asset Exposure (AE) dashboard.

Modified APIs

Modified APIs	Description
getReportApiData	Pagination support has been added to this API, along with new report filters applicationExclude (exclude specific applications) and patchType (filter by OS or third-party patches).

Depricated APIs

No APIs were deprecated in the 6.6 release.

Release 6.5

Newly Added API

No new APIs were added during the 6.5 release.

Modified APIs

Modified API	Description
addRemediationRule	A new key, **skipFeatureUpdate**, has been added to the API request. This optional boolean field allows you to specify whether feature updates should be considered when creating a remediation rule. • If set to true, feature updates will be skipped during remediation. • If set to false, feature updates will be included. If omitted, this will be the default behavior. The **skipFeatureUpdate** key only applies to the Saner Patch Management (PM) Remediation Rule.

updateRemediationRule	A new key, **skipFeatureUpdate**, has been added to the API request. This optional boolean field allows you to specify whether feature updates should be considered when creating a remediation rule. • If set to true, feature updates will be skipped during remediation. • If set to false, feature updates will be included. If omitted, this will be the default behavior. The **skipFeatureUpdate** key only applies to the Saner Patch Management (PM) Remediation Rule.
getDeviceJobInfo	A new field, '**tool'**, has been added to the '**getDeviceJobInfo**' API request. This field accepts values **CM** (Compliance Management) and **PM** (Patch Management) to let you retrieve job details specific to either module. If no value is provided, the API will return the asset's job details from Saner PM and CM by default. This is an optional field.

Deprecated APIs

No APIs were deprecated in the 6.5 release.

Release 6.4.1

Newly Added API

No new APIs were added during the 6.4.1 release.

Modified APIs

Modified API	Description
getBasicSystemDetails	○ a new key, **systemName,** has been added to the response. This field displays the hostname or the IP address assigned to the device.
getUser	○ a new key, **manageableOrganizationAndAccounts,** has been added to the response. This field displays the organization and the associated accounts of which the user is a part.

Deprecated APIs

No APIs were deprecated in the 6.4.1 release.

Release 6.4

Newly Added API

The following API is added to Saner Webservices.

New API	Description
getOrgWithAccountsHygieneScoreTrend	The **getOrgWithAccountsHygieneScoreTrend** fetches the scores for the last 30 days if available. If the organization doesn't contain the scores from the last 30 days, it will return them from when they were first calculated until the API was called. If the organization score is not calculated on some of the last 30 days, the most recently calculated scores according to those missing days will be displayed on those days.

Modified APIs

Modified APIs	Description
getAllBenchmarkProvisions	We have revamped the response for the **getAllBenchmarkProvisions** API. The new response contains the benchmark name, source account name, target account name, and target organization name. The getAllBenchmarkProvisions no longer requires adminID as a mandatory parameter in the request.

Deprecated API

We have deprecated the below API from the Saner Webservices Guide.

Deprecated API	Alternative API
getNonSecurityRemediationJobStatus	**getNonSecurityRemediationJobStatus** has been deprecated. Instead, you can use the **getRemediationJobStatus** API to get the status of the job/task and related info related to the non-security remediation job.
createNonSecurityRemediationJob	**createNonSecurityRemediationJob** has been deprecated. Instead, you can use the **createRemediationJob** to create a remediation job for missing security patches.
deleteNonSecurityRemediationJob	**deleteNonSecurityRemediationJob** has been deprecated. Instead, you can use the **DeleteRemediation** api to create a remediation job for missing security patches.

Release 6.3.1

Newly Added API

No new APIs were added during the 6.3.1 release.

Modified API

Modified API	Description
getDevice	○ a new key, **agentStatus,** has been added to the response. This field lets you know the current status of the Saner Agent on the device. ○ a new key, **agentConfig,** has been added to the response. This field lets you know the name of the Saner Agent profile assigned to the device. ○ a new key, **lastRemediation** has been added to the response. This field displays the date and time of the last remediation activity performed on the device.

Deprecated APIs

No APIs were deprecated in the 6.3.1 release.

Release 6.3.0.1

Newly Added APIs

No new APIs were introduced in the 6.3.0.1 release.

Modified APIs

No existing APIs were modified in the 6.3.0.1 release.

Deprecated APIs

We have deprecated the below API from the Saner Webservices Guide.

Deprecated API	Alternative API
getAllRemediations	**getAllRemediations** has been deprecated.

	Instead, you can use the **getapplicableremediation** API to fetch all the missing security patches by group, hostname,and family for an account.

Release 6.3

Newly Added APIs

No new APIs were introduced in the 6.3 release.

Modified APIs

Modified API	Description
addNetworkScanConfig	○ a new key, **enableudpports,** has been added to the request.
updateNetworkScanConfig	○ a new key, **enableudpports,** has been added to the request.
getDevice	○ a new key, **apifilters,** has been added to the request to fetch device information based on tags.
createRemediationJob	○ a new key, **patch_grouptype,** has been added to the request. Using the patch_grouptype filed, you can create a SECURITY or a NON-SECURITY remediation job.

Modified Report APIs

The 'getreportapidata' API has been modified to generate new reports. The newly introduced reports are listed below.

Report Name	Description
Microsoft Windows Vendor Patching Impact Graph	This graphical report provides information on the number of vulnerabilities that will be remediated after applying Microsoft Windows Vendor patches.
Third-Party Security Patching Impact Graph	This graphical report provides information on the number of vulnerabilities that will be remediated after applying third-party patches.
Installed Patches by Severity	This graphical report displays information on installed patches depending on their severity in a pie chart.
Network Interfaces -Mac	This report displays all the network interfaces on Mac systems present in the account.
Network Interfaces – Linux	This report displays all the network interfaces on Linux systems present in the account.

Modified Report APIs	Description
Vulnerabilities By Devices	Three new columns, namely, **Group, Ports/Service,** and **Description** have been added to the report.

Benchmark Deviation By Devices	Four new columns, namely, **Total Tules, Rules Passed, Rules Failed,** and **Non Compliant%** to the report.
Top 10 Vulnerable Hosts	a new column – '**IP Address'** has been added to the report.
Vulnerabilities By Devices	a new column – '**IP Address'** has been added to the report.
Newly Added Devices	a new column – '**IP Address'** has been added to the report.
Not Scanned Devices	a new column – '**IP Address'** has been added to the report.
Devices with Misconfigurations	a new column – '**IP Address'** has been added to the report.
Installed Misconfigurations Fixes By Devices	a new column – '**IP Address'** has been added to the report.
Misconfiguration Fix Details By Host Name	a new column – '**IP Address'** has been added to the report.
All applicable PA Reports	a new column – '**IP Address'** has been added to the report.
Patch Job Status	a new column – '**Created By'** has been added to the report.
Job Status Summary	a new column – '**Created By'** has been added to the report.
Patch Rollback Job Status	a new column – '**Created By'** has been added to the report.
Test and Deploy Patch Job Status	a new column – '**Created By'** has been added to the report.
Remediation Rule Status	a new column – '**Created By'** has been added to the report.
Installed Patches	a new column – '**Created By'** has been added to the report.
Installed Patches By Devices	a new column – '**Created By'** has been added to the report.
Misconfiguration Fixes Job Status	a new column – '**Created By'** has been added to the report.
Rollback Misconfiguration Fixes Job Status	a new column – '**Created By'** has been added to the report.
Firmware Job Status	a new column – '**Created By'** has been added to the report.

Deprecated APIs

No APIs were deprecated in the 6.3 release.

Release 6.2.1

Newly Added APIs

New API	Description
getDeviceVulnerabilities	The getDeviceVulnerabilities API returns all the vulnerabilities that exist on a device. You can choose if you want to list the vulnerabilities of a network device, endpoint device, or all the devices.
getAutomationRuleStatus	The getAutomationRuleStatus API returns the status of the automation rule in Saner PM (Patch Management) and CM (Compliance Management).
updateAutomationRuleStatus	The updateAutomationRuleStatus API pauses an active remediation rule in Saner PM (Patch Management) and CM (Compliance Management). At the same time, the 'updateautomationrule' API also resumes a remediation rule which is in pause status.

Modified APIs

Modified API	Description
createNonSecurityRemediationJob	○ a new key, **remjobpatchesinfo,** has been added to the request.
createFirmwareRemediationJob	○ a new key, **remjobpatchesinfo,** has been added to the request.
getApplicableRemediationJob	○ a new key, **patchinfo,** has been added to the response.

Deprecated APIs

No APIs were deprecated in the 6.2.1 release.

Release 6.2

Newly Added APIs

New API	Description
getAllRemediations	This method allows you to list all the missing security patches for an account by family, host, application, and severity. The response is grouped by vulnerable assets and provides detailed information on each patch and impacted hosts.

Modified APIs

Modified API	Description

getBasicSystemDetails	o a new key, **sanerEnabled,** has been added to the response. o a new key,**lastLogonUsername,** has been added to the response. o a new key, **deviceType,** has been added to the response**.**
getDeviceDetails	The API's response has been modified to provide the vulnerabilities and misconfigurations details in two separate JSON.

Deprecated API

No APIs were deprecated in the 6.2 release.

Release 6.1.1

Newly Added APIs

The following API has been added to Saner Webservices.

New API	Description
getDeviceTagKeys	This method will return all the tags assigned to devices. Using this API, you can fetch tags assigned to individual devices and all the devices in an account.

Modified API

The 'getreportapidata' API has been modified to generate new reports. The newly introduced reports are listed below.

Report Name	Description
Installed Patches by Device	This report provides a detailed device-wise view of the installed patches.
Missing Patches by Device	This report provides a detailed device-wise view of the missing patches.
Patch Compliance by Device	This report provides a detailed device-wise view of the patch compliance for the devices in the Account.

Deprecated API

No APIs were deprecated in the 6.1.1 release.

Release 6.1

Newly Added APIs

New APIs	Description
getBasicSystemDetails	This method lets you get the basic system details of the devices in an Account.
getInstalledApplications	This method gets all the installed applications on the devices in an Account.
getNonSecJobDetails	This method fetches the details of non-security remediation, firmware, rollback, and reboot jobs in Saner CM.
getCHScoreSummaryForDeviceWithStatus	This method fetches the cyber hygiene score, Saner Agent status, and hostnames for the devices found in the account.

Modified APIs

Modified APIs	Description
getDownloadUrl	existing key, **name,** has been changed to **account-name.**
updateUser	a new key, **UserGroup,** has been added to the request.
addUser	a new key, **UserGroup,** has been added to the request.
getReportApiData	a new key, **apifilters,** has been added to the request.
addDevice	a new key, **apifilters,** has been added to the request.
provisionBenchmark	a new key, **apifilters,** has been added to the request.
addRemediationRule	a new key, **apifilters,** has been added to the request.
getAllApplicableRules	a new key, **apifilters,** has been added to the request.
updateRemediationRule	a new key, **apifilters,** has been added to the request.
addSoftwareProvision	a new key, **apifilters,** has been added to the request.
updateDevice	a new key, **apifilters,** has been added to the request.
getDeviceDetails	a new key, **apifilters,** has been added to the request.
getDevicePdfReport	a new key, **apifilters,** has been added to the request.
Changes made to existing Risk Prioritization APIs.	a new key, **apifilters,** has been added to the request.
	Existing key,**user id,** has been removed from the request. Earlier, the key 'user id' was used to fetch

	the user details. Now, we are retrieving user details from the session id itself.

Deprecated API

We have deprecated the below API from the Saner Webservices Guide.

Deprecated API	Alternative API
getChScoreForDevice	**getChScoreSummaryForDevice** has been deprecated. Instead, you can use the **getCHScoreSummaryForDeviceWithStatus** API to fetch the score for the devices in an Account and the status of the Saner Agent.

Release 6.0

Newly Added APIs

The following APIs were added to Saner webservices.

New APIs	Description
getStatus	This method lets you get the basic system details of the devices in an Account.
getPostureAnomaly	This method gets all the installed applications on the devices in an Account.
getConfiguration	This method fetches the details of non-security remediation, firmware, rollback, and reboot jobs in Saner CM.
getConfigurationStatus	This method fetches the cyber hygiene score, Saner Agent status, and hostnames for the devices found in the account.
getWhitelist	This method is used to fetch fully whitelisted PA-ID's for a given account.
getAllConfiguration	This method is used to get all the configurable entries for a given account.
getDeviceDetails	This method is used to get a given device's vulnerability and compliance details.
getBasicSystemDetails	This method lets you get the basic system details of the devices in an Account. Details such as the IP address, MAC address, memory, disk size, serial number, systemUUID, CPU information, and operating system details are provided in the response when the 'getBasicSystemDetails' API is executed.
getInstallledApplications	This method gets all the installed applications on the devices that exist in an Account. Details such as the application name, version, publisher, and host details are returned as a response when the 'getInstalledApplications' API is executed.

getMisconfigurationRollbackStatus	This method gets the status of the job/task and related info related to the misconfiguration rollback status job.
deleteMisconfigurationRollbackTask	This method gets the status of the job/task and related info related to the misconfiguration rollback status job.
getMissionCriticalDeviceData	This method provides the details of assets and devices configured as mission-critical.
saveMissionCritcalDeviceData	This method is used to save critical assets, critical device, and configured questions.
getRiskPrioritizationnSummary	This method provides the count of risks under the Act, Attend, Track*, and Track categories.
getRiskSummary	This method gives a summary of the risk associated with risk id
getRiskDetails	This method lists the risks based on Priority (Act/ Attend/ Track*/ Track), also includes risk information such as ID, Title, Affected Products, Affected Devices, Mission Prevalence, Exploitation Status, Automatable, Technical Impact, and Fix details.
getRiskMitigationDetails	This method gives tactics and techniques, count of act, attend, track, and track*, asset names, and affected devices for each mitigation id. Here, in this API, specifying 'row_limit:0' will fetch all the entries.
getCriticalAssetRisksDetails	This method is used to get a tabular listing of critical software assets based on Priority (Act/ Attend/ Track*/ Track) and affected device count. Here, in this API, specifying 'row_limit:0' will fetch all the entries.
getRiskAutomatabilityDetails	This method is used to get the details about a vulnerability – whether it is automatable or not?
getRiskTechnicalImpact Details	This method is used to get the technical impact of a vulnerability. Technical impact is similar to the Common Vulnerability Scoring System (CVSS) base score's concept of "severity." When evaluating technical impact, the definition of scope is particularly important.
getRiskExplotabilityDetails	This method is used to get the exploitability details of a vulnerability. Exploitation determines the present state of exploitation of the vulnerability. It does not predict future exploitation or measure feasibility or ease of adversary development of future exploit code; rather, it acknowledges available information at time of analysis.
getDeviceRiskDetails	This method returns a tabular listing of critical devices based on Priority (Act/Attend/Track*/Track), and it also includes device details such as Primary OP address and group. Here, in this API, specifying 'row_limit:0' will fetch all the entries.
getRPJson	This method is used to fetch all the information about risk in a CycloneDX format.
getRiskPrioritizationStatus	This method is used to get the last scan time and status of the RP Scan.
getRiskonMissionCritical	This method shows the percentage of essential devices for business mission prevalence that are at risk.
getRiskonMIssionPrevalence	This method gives a summary of risk count on devices based on Mission Prevalence.

getRiskonEssentialDevices	This method provides the summary of risk count essential devices that are categorized as Business Centric, Data Storage and Public Facing. Identifying such MEFs is part of business continuity planning and crisis planning
prioritizeRisks	This method is used to perform an RP Scan on an Account.
getRPTrends	This method is used to determine how many risks and affected devices fall into act, attend, track, and track* for the last 30 scans. Multiple scans performed during the day will be considered a single scan.
getChainableRisks	This method is used to get each device's list of chainable risks.
getMVEDetails	This method is used to get the details of the MVE.
getAccountHygieneScore	This method fetches the cyber hygiene score of the given account.
calculateHygieneScore	This method allows you to calculate the Hygiene Score for the given account.
getCHScoreSummaryForFamily	This method returns the cyber hygiene score, hostnames, IP addresses, local score, global score, and OS Group per family.
getCHScoreSummaryForOS	This method returns the cyber hygiene score, hostnames, IP addresses, local score, global score, family, and group per OS.
getCHScoreSummaryForGroup	This method returns the hygiene score, hostnames, IP addresses, local score, global score, family, and OS per group.
getTrendingCHScore	This GET method fetches the scores for the last 30 days if available. If the account doesn't contain the last 30 days scores, then it will return the scores from the time they were first calculated until the time the API was called.
getCHSFrequencyDistribution	This method returns the number of devices per score interval.
getTopCHSAttributesOfPA	This method returns the top five contributors of PA with hosts, name, and weightage.
getTopCHSAttrbutesOfVM	This method returns the top five contributors of VM with hosts, name, severity, and weightage.
getTopCHSAttributesOfCM	This method returns top five contributors of CM with hosts, name, severity, and weightage.
getTopCHSAttributesOfPM	This method returns top five contributors of PM with hosts, name, severity, and weightage
getOrgCHScoreByFamily	This method returns the CHS Score of the Organization based on the family.
getOrgCHScoreByOS	This method returns the CHS Score of the Organization based on the group.
getOrgCHScoreByOS	This method returns the CHS Score of the Organization based on the operating system
getOrgHygieneScore	This method returns the organization's score and score details of the accounts that are calculated and present in the given organization.
GetTopCHSAttributes	This method returns the top five anomalies with scores, ccs with scores, cvs with scores, missing patches with scores.
getCHSWeightage	This method returns the weights of anomaly, compliance, missing patches and vulnerabilities in calculation of cyber hygiene score.

updateCHSWeightage	This method is used to update the weights of anomaly, compliance, missing patches, vulnerabilities and starts a CHS Scan
getCHScanStatus	This method returns the CHS scan status of an Account.
updateOrgCHSWeightage	This method updates the weights of accounts under the given organization. Scans the accounts after updating.
getTrendingOrgHygieneScore	This method fetches the score of the last 30 days of the Organization if they are available.
getAssetsByVulnerability	This method is used to get the lists of assets linked to the CVE or CVEs specified in the reference value.
deleteRemediation	This method is used to delete any task/job created by using the 'createremediation' api.
getNonSecurityRemediationJobStatus	This method gets the status of the job/task and related info related to the non-security remediation job.
deleteNonSecurityRemediation	This method is used to delete a non-security job which is created as part of non-security remediation
getApplicableFirmwareRemediation	This method gets the list of firmware patches available for the account or the list of devices provided.
getFirmwareRemediationJobStatus	This method gets the status of the job/task created as part of the firmware remediation job.
deleteFirmwareRemediationJob	This method is used to delete any existing firmware remediation jobs.
getPatchRollbackStatus	This method is used to fetch the status of job/task and related info created as part of patch rollback status.
deletePatchRollbackTask	This method is used to delete any created patch rollback task.

Modified Report APIs

Modified Report APIs	Description
Get Report Apis	**start date** and **end date** filters have been added to search for a date range for the following reports: • Remediation Patch Details by Asset • Misconfiguration Fix Details by Asset • Remediation Patch Details by Task • Misconfiguration Fix Details by Task • Remediation Patch Details by Group • Misconfiguration Fix Details by Group • Remediation Patch Details by Host Name • Misconfiguration Fix Details by Host Name • Remediation Patch Summary • Misconfiguration Fix Summary
All Devices	Two new columns -'**Build Version**' and '**Family**' has been added to the report.
Device Details	Two new columns -'**Build Version**' and '**Family**' has been added to the report.
Patch Compliance Report	The columns below have been added to the report. • Vendor • Release Date • Detection Date

	• Patch age in days • Total Risk Count • Risks (References)
Patch Summary Report	A new column – '**Release Date'** has been added to the report.

Deprecated API

We have deprecated the below API from the Saner Webservices Guide.

Deprecated API	Alternative API
undeploysaner	**undeploysaner** has been deprecated. Instead, you can use the **uninstallagent** API to un-install the Saner Agent from the device.