Examples

CURL

cURL is a command-line tool for sending http/https requests and commands. It can be used to interact with the API. It is pre-installed on many Linux and Mac systems. Some examples:

GET

```
$ curl -u foo:bar -k https://127.0.0.1:55000
```

```
{"error":"0","data": "Welcome to Wazuh HIDS API"}
```

PUT

```
$ curl -u foo:bar -k -X PUT https://127.0.0.1:55000/agents/new_agent
```

```
{"error":0,"data":"004"}
```

POST

```
$ curl -u foo:bar -k -X POST -d '{"name":"NewHost","ip":"10.0.0.8"}' -H 'Content-Type:application/json'
"https://127.0.0.1:55000//agents"
```

```
{"error":0,"data":"004"}
```

DELETE

```
$ curl -u foo:bar -k -X DELETE https://127.0.0.1:55000/rootcheck/001
```

```
{"error":"0","data":"Policy and auditing database updated"}
```

Python

It is very easy to interact with the API using Python:

Code:

```python
#!/usr/bin/env python

import json
import requests # To install requests, use: pip install requests

# Configuration
base_url = 'https://IP:55000'
auth = requests.auth.HTTPBasicAuth('foo', 'bar')
verify = False
requests.packages.urllib3.disable_warnings()

# Request
url = '{0}{1}'.format(base_url, "/agents/000")
r = requests.get(url, auth=auth, params=None, verify=verify)
print(json.dumps(r.json(), indent=4, sort_keys=True))
print("Status: {0}".format(r.status_code))
```

Output:


```
{
    "error": "0",
    "data": {
        "id": "000",
        "ip": "127.0.0.1",
        "lastKeepAlive": "Not available",
        "name": "LinMV",
        "os": "Linux LinMV 3.16.0-4-amd64 #1 SMP Debian 3.16.7-ckt11-1 (2015-05-24) x86_64",
        "rootcheckEndTime": "Unknown",
        "rootcheckTime": "Unknown",
        "status": "Active",
        "syscheckEndTime": "Unknown",
        "syscheckTime": "Unknown",
        "version": "OSSEC HIDS v2.8"
    }
}
Status: 200
```


For a fuller example, see `/var/ossec/api/examples/api-client.py`.

PowerShell

The **Invoke-RestMethod** cmdlet sends requests to the API and handles the response easily. This cmdlet was introduced in Windows PowerShell 3.0.

Code:

```
function Ignore-SelfSignedCerts {
    add-type @"
        using System.Net;
        using System.Security.Cryptography.X509Certificates;

        public class PolicyCert : ICertificatePolicy {
            public PolicyCert() {}
            public bool CheckValidationResult(
                ServicePoint sPoint, X509Certificate cert,
                WebRequest wRequest, int certProb) {
                return true;
            }
        }
"@
    [System.Net.ServicePointManager]::CertificatePolicy = new-object PolicyCert
}

# Configuration
$base_url = "https://IP:55000"
$username = "foo"
$password = "bar"
$base64AuthInfo = [Convert]::ToBase64String([Text.Encoding]::ASCII.GetBytes(("{0}:{1}" -f $username,
$password)))
Ignore-SelfSignedCerts

# Request
$url = $base_url + "/syscheck/000/last_scan"
$method = "get"
try{
    $r = Invoke-RestMethod -Headers @{Authorization=("Basic {0}" -f $base64AuthInfo)} -Method $method -Uri
$url
}catch{
    $r = $_.Exception
}

Write-Output $r
```

Output:

```
error data
----- --------
0     @{syscheckTime=Wed Feb 24 09:55:04 2016; syscheckEndTime=Wed Feb 24 10:00:42 2016}
```

For a fuller example, see `/var/ossec/api/examples/api-client.ps1`.