CVE-2022-25165: Privilege Escalation to SYSTEM in AWS VPN Client

David Yesland

Vulnerabilities Overview

Affected Product

The AWS VPN Client application is affected by an arbitrary file write as SYSTEM, which can lead to privilege escalation and an information disclosure vulnerability that allows the user's Net-NTLMv2 hash to be leaked via a UNC path in a VPN configuration file. These vulnerabilities are confirmed to affect version 2.0.0 and have been fixed in version 3.0.0.

To fix the vulnerabilities, upgrade to version 3.0.0 which can be downloaded here. (https://aws.amazon.com/vpn/client-vpn-download/)

Vendor: Amazon Web Service (AWS)
Product: AWS VPN Client (Windows) (https://aws.amazon.com/vpn/client-vpn-download/)
Confirmed Vulnerable Version: 2.0.0
Fixed Version: 3.0.0

A race condition exists during the validation of OpenVPN configuration files. This allows OpenVPN configuration directives outside of the AWS VPN Client allowed OpenVPN directives list (https://docs.aws.amazon.com/vpn/latest/clientvpn-user/connect-aws-client-vpn-connect.html) to be injected into the configuration file prior to the AWS VPN Client service, which runs as SYSTEM, processing the file. Dangerous arguments can be injected by a low-level user such as “log” which allows an arbitrary destination to be specified for writing log files.

The impact is an arbitrary file write as SYSTEM with partial control over the contents of the file. This can lead to local privilege escalation or denial of service.

CVE-2022-25165: Information Disclosure via UNC Path

It is possible to include a UNC path in the OpenVPN configuration file when referencing file paths for directives (such as “auth-user-pass”). When this file is imported to the AWS VPN Client and the client attempts to validate the file path it performs an open operation on the path and leaks the user’s Net-NTLMv2 hash to an external server.

The impact is information leakage of a user’s Net-NTLMv2 hash. This could be exploited by having a user attempt to import a malicious VPN configuration file into the AWS VPN Client.

What Is AWS VPN Client

AWS VPN Client is a desktop application that can be used to connect to the AWS Client VPN.

From the product website:

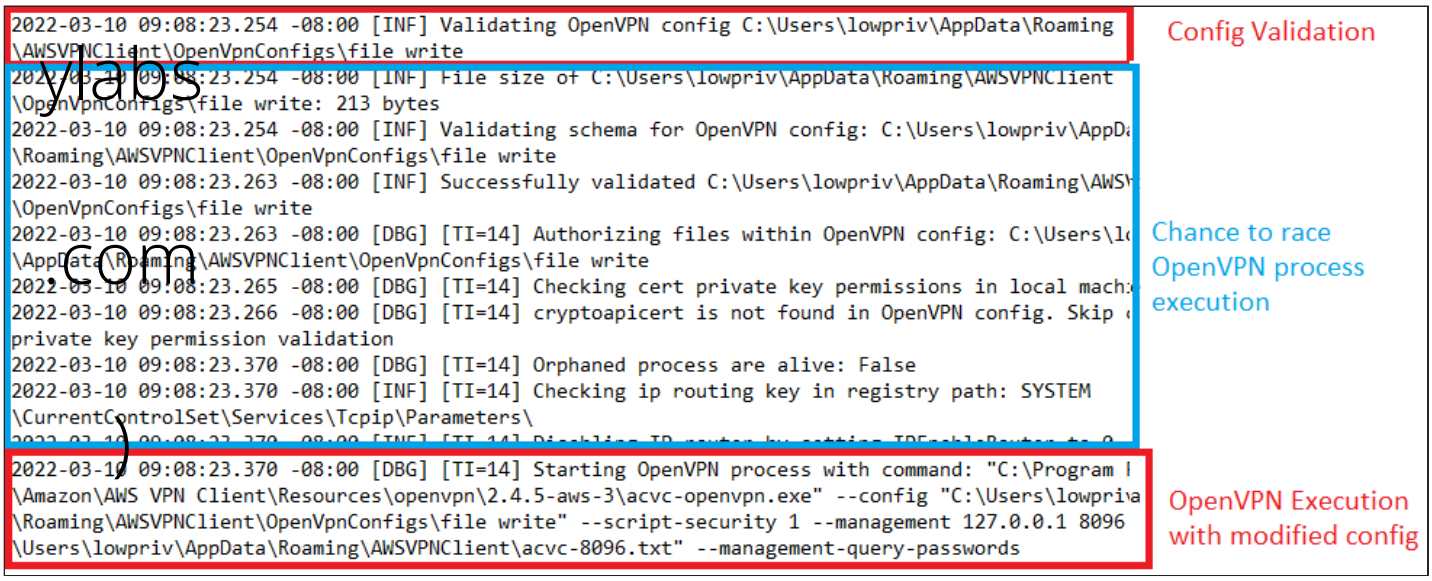
The client for AWS Client VPN is provided free of charge. You can connect your computer directly to AWS Client VPN for an end-to-end VPN experience. The software client is compatible with all features of AWS Client VPN.

Arbitrary File Write as SYSTEM Technical Details

AWS VPN Client installs a Windows service which runs as SYSTEM acting as a wrapper to a custom OpenVPN client executable. A low privileged user can use the AWS VPN Client to attempt to connect to a VPN using an imported OpenVPN configuration file.

There are known dangerous OpenVPN directives (https://openvpn.net/community-resources/reference-manual-for-openvpn-2-4/#scripting-and-environmental-variables) which perform actions such as running commands or writing log files to a specific destination during a VPN connection. AWS VPN Client attempts to restrict the OpenVPN directives which can be used (https://docs.aws.amazon.com/vpn/latest/clientvpn-user/connect-aws-client-vpn-connect.html) in the configuration file, but the check fails as it is performed prior to the execution of the OpenVPN executable. This makes it possible to race the execution of the OpenVPN executable after the configuration file has been validated and inject disallowed directives into the file.

Below you can see a log file produced by the AWS VPN service which shows the time between the successful validation of the configuration and the execution of the OpenVPN client.



It is easy enough to use a Powershell script to then monitor the log file and upon successful validation of the configuration file immediately write the malicious directive to the configuration file prior to the OpenVPN executable processing it.

With the ability to inject disallowed directives it would seem as easy as adding one of the directives which allows executing a command into the configuration file giving an easy path to privilege escalation. Although in this case it was not this straightforward as

(888) 944-8679 (TEL:1-888-944-8679)
CONTACT US (HTTPS://RHINOSECURITYLABS.COM/CONTACT/)
GET A QUOTE
(HTTPS://RHINOSECURITYLABS.COM/LANDING/REQUEST-A-QUOTE/)
ASSESSMENT SERVICES (HTTPS://RHINOSECURITYLABS.COM/SERVICES/)
INDUSTRIES (HTTPS://RHINOSECURITYLABS.COM/INDUSTRY/)
RESOURCES (HTTPS://RHINOSECURITYLABS.COM/RESOURCES/)
SECURITY BLOG (HTTPS://RHINOSECURITYLABS.COM/BLOG/)
COMPANY (HTTPS://RHINOSECURITYLABS.COM/COMPANY/)

AWS VPN service starts the OpenVPN executable with the "–script-security 1" flag, which prevents external binaries or scripts from being executed.

Although we cannot directly run commands, it is still possible to use the `log` directive to redirect log output to any path or file of our choosing. Since execution is done as the SYSTEM user this gives us a privileged file write where we partially control the content. In the simplest case this could be used to write a batch script to an administrator's startup directory.

A proof of concept for CVE-2022-25166 can be found on our Github repo here. (https://github.com/RhinoSecurityLabs/CVEs/tree/master/CVE-2022-25166)

```
Windows PowerShell
Group Name                             Type             SID          Attributes
====================================== ================ ============ ==================================================
Everyone                               Well-known group S-1-1-0      Mandatory group, Enabled by default, Enabled group
BUILTIN\Users                          Alias            S-1-5-32-545 Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\INTERACTIVE               Well-known group S-1-5-4      Mandatory group, Enabled by default, Enabled group
CONSOLE LOGON                          Well-known group S-1-2-1      Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Authenticated Users       Well-known group S-1-5-11     Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\This Organization         Well-known group S-1-5-15     Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\Local account             Well-known group S-1-5-113    Mandatory group, Enabled by default, Enabled group
LOCAL                                  Well-known group S-1-2-0      Mandatory group, Enabled by default, Enabled group
NT AUTHORITY\NTLM Authentication       Well-known group S-1-5-64-10  Mandatory group, Enabled by default, Enabled group
Mandatory Label\Medium Mandatory Level Label            S-1-16-8192


PRIVILEGES INFORMATION
----------------------

Privilege Name                Description                          State
============================= ==================================== ========
SeShutdownPrivilege           Shut down the system                 Disabled
SeChangeNotifyPrivilege       Bypass traverse checking             Enabled
SeUndockPrivilege             Remove computer from docking station Disabled
SeIncreaseWorkingSetPrivilege Increase a process working set       Disabled
SeTimeZonePrivilege           Change the time zone                 Disabled

PS C:\Users\lowpriv>
```

Information Disclosure via UNC Path Technical Details

AWS VPN Client performs validation on configuration files which are imported into the client as a VPN profile. One of the validation steps consists of checking if a file path exists when any file paths are supplied to directives which accept a file path as an argument.

Some examples of valid directives that accept files paths are

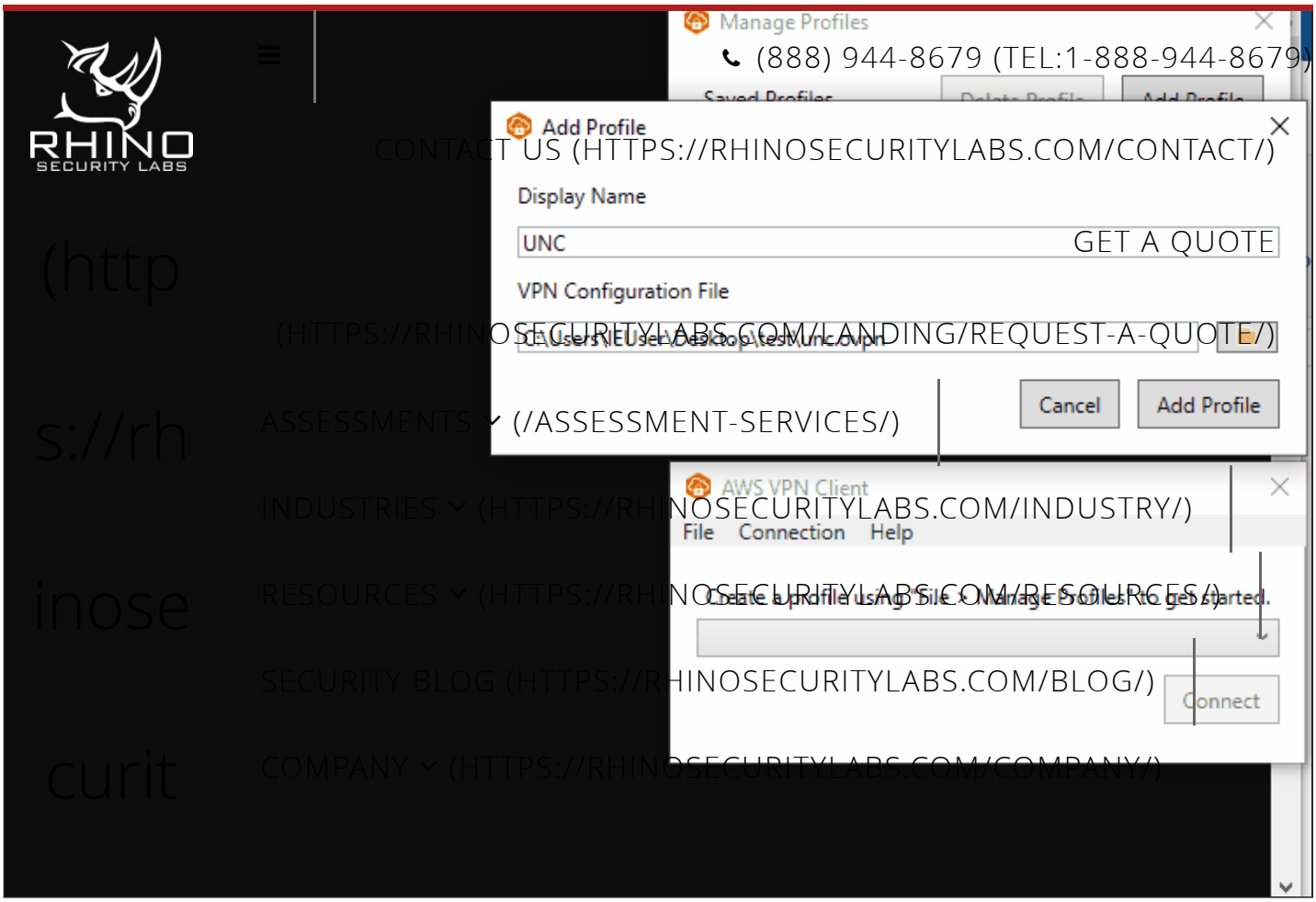
- auth-user-pass
- ca

Validation by AWS VPN Client is done by performing a file open operation on the path to ensure it exists.

```
private static void CheckFilePath(Regex regex, string line)
{
    string text = regex.Match(line).Groups[1].Value.Trim();
    string text2 = regex.Match(line).Groups[2].Value.Trim().Replace("\"", "").Rep
    Log.Information("Validating " + text + " path: " + text2);
    try
    {
        using (FileStream fileStream = File.Open(text2, FileMode.Open))
        {
            fileStream.Close();
        }
```

AWSVPNClient.Core.dll contains OvpnConfigParser.cs which has the “CheckFilePath” method used to check if a file path is valid. This simply calls File.Open on the supplied file name.

This can be exploited by providing a file that contains UNC paths as the file path. When the file is validated before being imported it will open the UNC path and send the user’s Net-NTLMv2 hash to an external server.



A proof of concept for CVE-2022-25165 can be found on our Github repo here. (https://github.com/RhinoSecurityLabs/CVEs/tree/master/CVE-2022-25165)

Conclusion

Disclosure Timeline

Similar to the Pritunl CVE and blog post (https://rhinosecuritylabs.com/penetration-testing/cve-2022-25372-local-privilege-escalation-in-pritunl-vpn-client/) recently released, this vulnerability demonstrates the exploit potential in high-privilege Windows processes — such as those used by VPN clients. This also shows how common application flaws are still present in sensitive applications, whether from open source providers or major tech companies.
Stay tuned for another VPN release in the coming weeks!

2/15/2022	Vulnerabilities reported to AWS
2/16/2022	Acknowledged by AWS
3/4/2022	AWS confirmed the issues have been addressed in version 3.0.0
4/13/2022	Full Disclosure (blog post) released

Related Resources

(https://rhinosecuritylabs.com/aws/cve-2021-38112-aws-workspaces-rce/)

CVE-2021-38112:
AWS WorkSpaces Remote Code Execution

(https://rhinosecuritylabs.com/aws/cloud-malware-cloudformation-injection/)

Cloud Malware:
Resource Injection in CloudFormation Templates

(https://rhinosecuritylabs.com/aws/exploring-aws-ebs-snapshots/)

Downloading and Exploring AWS EBS Snapshots

Interested in more information?

20603

Contact Us Today »

(888) 944-8679 (TEL:1-888-944-8679)

CONTACT US (HTTPS://RHINOSECURITYLABS.COM/CONTACT/)

GET A QUOTE (HTTPS://RHINOSECURITYLABS.COM/LANDING/REQUEST-A-QUOTE/)

ASSESSMENTS (/ASSESSMENT-SERVICES/)

INDUSTRIES (HTTPS://RHINOSECURITYLABS.COM/INDUSTRY/)

RESOURCES (HTTPS://RHINOSECURITYLABS.COM/RESOURCES/)

SECURITY BLOG (HTTPS://RHINOSECURITYLABS.COM/BLOG/)

COMPANY (HTTPS://RHINOSECURITYLABS.COM/COMPANY/)

ASSESSMENT SERVICES (HTTPS://RHINOSECURITYLABS.COM/ASSESSMENT-SERVICES/)

Network Penetration Test (https://rhinosecuritylabs.com/assessment-services/network-penetration-testing/)

Webapp Penetration Test (https://rhinosecuritylabs.com/assessment-services/web-penetration-testing/)

AWS Cloud Penetration Testing (https://rhinosecuritylabs.com/assessment-services/aws-cloud-penetration-testing/)

GCP Cloud Penetration Testing (https://rhinosecuritylabs.com/assessment-services/gcp-penetration-testing/)

Azure Penetration Testing (https://rhinosecuritylabs.com/assessment-services/azure-penetration-testing/)

Mobile App Assessment (https://rhinosecuritylabs.com/assessment-services/mobile-app-assessment/)

Secure Code Review (https://rhinosecuritylabs.com/assessment-services/secure-code-review/)

Social Engineering / Phishing Testing (https://rhinosecuritylabs.com/assessment-services/social-engineering/)

Vishing (Voice Call) Testing (https://rhinosecuritylabs.com/assessment-services/social-engineering/vishing-assessments/)

Red Team Engagements (https://rhinosecuritylabs.com/assessment-services/red-team-engagement/)

INDUSTRIES (HTTPS://RHINOSECURITYLABS.COM/INDUSTRY/)

Healthcare (https://rhinosecuritylabs.com/industry/healthcare/)

Finance (https://rhinosecuritylabs.com/industry/financial/)

Technology (https://rhinosecuritylabs.com/industry/technology/)

Retail (https://rhinosecuritylabs.com/industry/retail/)

RESOURCES (HTTPS://RHINOSECURITYLABS.COM/RESOURCES/)

Technical Blog (https://rhinosecuritylabs.com/blog-technical/)

Strategic Blog (https://rhinosecuritylabs.com/blog-strategic/)

Example Pentest Report (https://rhinosecuritylabs.com/landing/penetration-test-report/)

Technical Research (https://rhinosecuritylabs.com/research-and-vulnerability-disclosure/)

Vulnerability Disclosures (https://rhinosecuritylabs.com/research-and-vulnerability-disclosure/)

Disclosure Policy (https://rhinosecuritylabs.com/company/vulnerability-disclosure-policy/)

Penetration Testing FAQ (https://rhinosecuritylabs.com/assessment-services/penetration-testing-faq/)

Support: AWS Pentest Form (https://rhinosecuritylabs.com/assessment-services/support-aws-penetration-testing-form/)

()

COMPANY (HTTPS://RHINOSECURITYLABS.COM/COMPANY/)

Leadership (https://rhinosecuritylabs.com/company/leadership/)

Blog (https://rhinosecuritylabs.com/blog/)

Careers (https://rhinosecuritylabs.com/careers/)

Company Principles (https://rhinosecuritylabs.com/careers/rhino-company-principles/)

Contact Us (https://rhinosecuritylabs.com/contact/)

Get a Quote (https://rhinosecuritylabs.com/request-a-quote/)

RSS Feed (https://rhinosecuritylabs.com/blog/feed/)

ABOUT US

Rhino Security Labs is a top penetration testing and security assessment firm, with a focus on cloud pentesting (AWS, GCP, Azure), network pentesting, web application pentesting, and phishing.

With manual, deep-dive engagements, we identify security vulnerabilities which put clients at risk.

Endorsed by industry leaders, Rhino Security Labs is a trusted security advisor to the Fortune 500.

(https://rhinosecuritylabs.com/)

info@rhinosecuritylabs.com (mailto:info@rhinosecuritylabs.com)

(888) 944-8679 (tel:1-888-944-8679)

Rhino Security Labs, Inc