FIELD NOTES FROM A PRESALES SOLUTIONS ARCHITECT

Dozens of VPN Credentials in a Leak — What to Do in the First 24 Hours

A field note about a threat report that said "no evidence of valid credentials" — the one sentence almost nobody challenges — and the order of work I now put on the table the moment a VPN domain shows up in a leak.

Daanial Farrukh · 07 November 2026 · https://quabin.com/vpn-credentials-leak-first-24-hours/

I was reading a threat-intelligence appraisal.

It was a careful document. Sources, dates, a clean summary. It covered one of the large credential leaks of this year and it ended with a calm conclusion: "no evidence of valid or active credentials."

That sentence is the one people forward to their managers.

So I asked one question. Has anybody searched for the domain?

I searched. The lookup took less than a minute.

The organisation's own domain was there — dozens of SSL-VPN accounts, listed against its own edge firewalls. Not a staff member reusing a password on a shopping site. Real VPN portals, on the organisation's own public ranges, several on non-standard ports and a few on plain HTTP.

"No evidence" had only meant nobody had looked.

That's the trap. A leak doesn't show up as an alert. Your firewall is healthy, your patch level is fine, your monitoring is green — and someone else is holding the password.

What a Credential Leak Actually Is

A credential leak is not a vulnerability. It is a list. Someone already has the username and the password, and the attacker's next step is an ordinary login.

The public example this year was FortiBleed. Researchers reported a dataset covering about 74,000 FortiGate firewall URLs across 194 countries, and public analysis confirmed that sampled logins were real. Fortinet's position is that this was not a new vulnerability — the data came from earlier incidents and from brute-force attacks. That detail matters more than the headline number.

It means the path looks like this:

Old theft → dataset → password spray or direct login → a normal-looking VPN session

Nothing in that flow is an exploit. Your IPS has nothing to match. The login succeeds, and the log line looks exactly like Monday morning.

This pattern is common in three places:

- SSL-VPN portals with local accounts and no MFA
- Firewall management pages that were left open to the internet
- Directory bind accounts stored in the firewall configuration, which rarely change

Why the Usual Response Misses It

The usual response is one word: patch.

Patching is necessary. It is not sufficient. Here's why.

A patch closes a door. It does not change a key that was copied last year. If the password was taken before the patch, it still works after the patch. Public guidance on this leak says the same thing: if your domain or IP appears, assume compromise, and treat the upgrade as one step among several.

There is a second reason. Fortinet documented an attacker technique where a symbolic link was planted in the folder that serves SSL-VPN language files, and that link could survive a firmware upgrade. So an upgrade can leave you with a patched box that is still not clean.

And there is a third reason, the one that surprised me most. **The date of the leak is not the date of the theft.** In the case I was reading, the most likely source was not the firewall at all. It was a Windows server on the same network — logged in daily as the built-in Administrator, no antivirus, passwords saved in the browser. An infostealer had taken about a hundred saved logins, a handful of them corporate, years before the leak made the news.

That is still a theory until the cleartext list is compared, and I'd rather say so. But it changes the question. "Did we change our passwords after the news?" is the wrong test. The right test is: **was this password changed after the date it was stolen?**

How to Actually Fix This — The First 24 Hours

This is the order I recommend, and the order matters. The commands are FortiOS because that was the case in front of me. The sequence works on any VPN gateway.

Commands marked as read-only change nothing. Run those first.

Step 1: Hour 0 — confirm, then freeze the evidence. (Read-only.)

Search your domains and public IPs in a reputable lookup, such as the public checker Hudson Rock published for this dataset. Then take a configuration backup and export the logs before you change anything. In the next steps you will be removing the very things you want to study.

Record the firmware on every listed device:

```
get system status
```

Plain takeaway: a listing is a reason to start, not proof of a breach. It is enough to act on.

Step 2: Hour 1 — see who is connected, then cut them off.

List the active SSL-VPN users. (Read-only.)

```
get vpn ssl monitor
```

Read every line. Look for a source IP you can't explain, an account of someone on leave, a service account with an interactive session. Then remove the tunnels you do not trust. (This changes state.)

```
execute vpn sslvpn del-tunnel <index>
```

Cut sessions first and rotate second. A live session does not always care that the password changed underneath it.

Step 3: Hours 1–6 — rotate everything the firewall trusts. (This changes state.)

Most teams change the VPN user passwords and stop. That is about a third of the job. The real list is:

1. Local administrator accounts on every firewall
2. Local SSL-VPN users
3. LDAP and RADIUS bind accounts and shared secrets
4. API tokens and SNMP strings
5. Any password that was reused anywhere else

Item 3 is the one that hurts. A directory bind account stored in a firewall configuration is often years old, rarely reviewed, and very useful to an attacker. Treat the whole configuration as disclosed.

Check the bind account's age on the directory side. (Read-only.)

```
Get-ADUser <vpn-bind-account> -Properties PasswordLastSet,LastLogonDate
```

If PasswordLastSet is older than the suspected theft, that password is still in play — no matter what the news cycle says.

One more detail from the public advice: have each administrator log in after the rotation so the firewall can re-hash the password with PBKDF2. Older hashes are the ones that crack.

Step 4: Hours 2–12 — hunt for what the login left behind. (Read-only.)

Rotation locks out the password. It does not remove an account the attacker created with it.

List every administrator:

```
show system admin
```

List every local VPN user:

```
show user local
```

List the local-in policies and the automation stitches:

```
show firewall local-in-policy
```

```
show system automation-stitch
```

Fortinet's published incident write-ups name accounts like audit, backup, itadmin, secadmin and support as ones attackers have created. A plausible name is the whole point. Do not ask whether the name looks suspicious. Ask who created it and on what date.

Then read the login history. Filter the event log for logins:

```
execute log filter category 1
```

```
execute log filter field action login

execute log display
```

You are looking for a source that has never appeared before, a success after a burst of failures, and a dormant account that suddenly works. Then search the same window for configuration downloads and for changes to trusted hosts, MFA, logging or DNS.

Here's the part that trips people up. Log memory on a small firewall is short. If logs are not forwarded off the box, "nothing in the logs" may only mean the logs have rolled over. That is the appraisal's mistake again.

If you find a rogue account or an unexplained change, stop cleaning. Restore a known-clean configuration or rebuild the device, and escalate it as an incident.

Step 5: Hours 6–18 — close the door the password opened. (This changes state.)

First, patch to a current release. Then remove the exposure:

- Take firewall management off the internet. Researcher Kevin Beaumont noted in his analysis of this leak that management was exposed on most of the affected devices.
- Restrict each administrator to known sources:

```
config system admin
edit <admin-name>
set trusthost1 203.0.113.10 255.255.255.255
end
```

- Enforce MFA on every admin and VPN account, with no exceptions for service desks or "just this one director".
- Force HTTPS. A VPN portal on plain HTTP sends its login in the clear.

This is the same pattern as any admin lockdown: the port should answer only the people who need it.

Step 6: Hours 12–24 — find the machine the passwords came from.

This is the step everyone skips, because the firewall team has gone home feeling finished.

If the source was an infostealer, the firewall is only where the stolen password was used. The infected host has other secrets. Ask four questions:

1. Which machine had these logins saved in a browser?
2. Is it still in service? If so, reimage it. Do not clean it.
3. Is it used daily as Administrator? Stop that.
4. Does it run endpoint protection that someone actually watches?

Then widen the rotation to every login on that machine — mail, control panels, personal accounts that share a pattern. A password scheme with one changing word is one password.

Step 7: Verify — the step that proves any of this happened.

This step is where most tutorials stop and where mine starts.

Check three things. First, list the bind account again:

```
Get-ADUser <vpn-bind-account> -Properties PasswordLastSet
```

The date must be today's. Second, run show system admin and match every line to a named person. Third, from outside your network, try to reach the management page. It must time out.

```
curl -k -m 10 https://203.0.113.20:8443/
```

Expected result: a timeout. If you get a login page, step 5 didn't take.

Then do the check nobody enjoys. Have one user try to connect with a password but without the second factor. It must fail.

A refused login here is not a fault. It is the verification.

The Bigger Lesson

Three things from this case have stayed with me.

"No evidence" is not a finding. It is a statement about how hard you looked. Before you accept it, ask what was searched, and for which domains.

A patch is a date, not a state. It tells you when the door closed. It says nothing about who already has a key.

The firewall is rarely where the story begins. One unprotected server used for daily work can feed a VPN, a mail system and a control panel at the same time.

The Checklist I Run Now

1. Domains and public IPs searched in a leak lookup — by a person, with a date
2. Active sessions listed and untrusted ones cut: get vpn ssl monitor
3. Admin, VPN, bind, RADIUS, API and SNMP secrets rotated, not just user passwords
4. PasswordLastSet newer than the suspected theft date
5. Every administrator matched to a named person: show system admin
6. Management closed from outside, proven with curl from an outside network
7. MFA proven by a failed login without the second factor
8. Logs forwarded off the box, so the next review has a history to read

Plain takeaway: rotate first, hunt second, and do not close the ticket until you have found the source.

What I Tell Every Customer Now

When a report lands saying the organisation is not affected, I ask one question:

"Who searched for our domain in the dataset, on what date, and which of our passwords are older than the theft?"

If the answer is a name, a date and a list, they're in good shape. If the answer is "the report says we're fine", we're not done yet.

I write about the questions I ask in real customer rooms — the ones that change architectures and surface the problems no one else noticed. If you found this useful, drop your email below and I'll send the next one when it's ready.

FAQs

What should I do first if my VPN credentials appear in a leak?

Confirm the listing, back up the configuration and export the logs, then list active VPN sessions and cut any you cannot explain. Do this before rotating passwords. Assume the listed accounts are compromised until the passwords have been changed.

Does patching the firewall fix a credential leak?

No. A patch closes a software flaw, but a stolen password still works after the upgrade. You must also rotate credentials, check for rogue accounts and enforce MFA. Fortinet has also documented an attacker technique that could survive a firmware upgrade, so a patched box is not automatically clean.

Which passwords need to change after a firewall credential leak?

All of them that the firewall holds or trusts: local administrators, local VPN users, LDAP and RADIUS bind accounts, shared secrets, API tokens and SNMP strings. Also change any password reused elsewhere. Treat the whole configuration as disclosed, because a backup of it may have been taken.

How do I know if a leaked VPN password was actually used?

Read the login history for new source addresses, successes after a burst of failures and dormant accounts that suddenly log in. Then check for new administrators, changed trusted hosts and configuration downloads. If logs were not forwarded off the firewall, an empty result proves little.

Why would old passwords appear in a leak from this year?

The date a dataset is published is not the date the passwords were stolen. An infostealer on one machine can capture saved browser logins years before they appear in a public dataset. That is why the useful test is whether a password was changed after the theft, not after the news.

Is it enough to enable MFA after a VPN credential leak?

It is one of the strongest steps, but it does not replace rotation. Cut existing sessions, rotate the passwords and look for rogue accounts as well, because an attacker who already got in may have created a way back. Then prove MFA works by trying a login without the second factor.

Read online: https://quabin.com/vpn-credentials-leak-first-24-hours/