contact@namiq.io · namiq.io

NamiQ

SECURITY ADVISORY

nginx2root

CVE-2026-123007 and four more · nginx 1.9.5 to 1.31.8

Assessment & Action Plan

nginx 1.9.5 to 1.31.8 with HTTP/2 on is open to CVE-2026-123007, an unauthenticated memory corruption, unless a backport fixed it. Four more need set-ups.

PREPARED BY	Hung Le, NamiQ	DATE	19 November 2026
FACTS AS OF	19 November 2026	VERSION	1.0
SEVERITY	**Critical: CVSS 4.0 9.2, CVSS 3.1 8.1 (F5, all five)**	STATUS	Fixed in 1.31.9 and 1.30.7; not on CISA's exploited list; CISA: exploitation none

NamiQ © 2026 NamiQ. All rights reserved. This document is the property of NamiQ. It may be shared unaltered, with attribution.
Companion article: namiq.io/insights/nginx2root-cve-2026-123007-http2

At 15:54 UTC on 18 November 2026 nginx released 1.31.9, and two minutes later 1.30.7, with fixes for ten vulnerabilities. Five of them were reported by Xint, which published its research the same day under the name nginx2root. F5, which maintains nginx, published a CVE for each and scored all five alike, CVSS 3.1 8.1 and CVSS 4.0 9.2, though nginx.org rates two of them major (CVE-2026-123007 and CVE-2026-113425) and three medium.

The one that matters to almost everyone is CVE-2026-123007, a memory-corruption bug in how nginx counts HTTP/2 flow-control windows. It needs no account and no special configuration beyond HTTP/2 being turned on, and it is in every nginx release from 1.9.5 to 1.31.8, 1.9.5 being the version that added HTTP/2 in 2015. Xint demonstrated running commands on the server with it, on AWS, Google Cloud and Azure, from inside the same network and over a public address. The other four need a particular set-up, and for most servers one or two of them apply at most.

The name overstates one thing. The nginx bugs give an attacker the rights of an nginx worker process, not root. Xint reached root by chaining a sixth bug, in the Linux kernel's SCTP code, which has no CVE and no published patch yet. This note is for whoever runs nginx, directly or inside something else: which of the five reach your servers, what to do in what order, and the one deployment that cannot get a fix from its own project at all. Everything below is as of 19 November 2026, and every claim links to its source at the end.

The short version

- Update to nginx 1.31.9 (mainline) or 1.30.7 (stable), or your distribution's fixed package, and restart nginx. A reload does not replace the running program.
- CVE-2026-123007 needs only HTTP/2 enabled. If nginx speaks HTTP/2 to the internet, this is today's job.
- Turning HTTP/2 off blocks two of the five (CVE-2026-123007 and CVE-2026-123633). The other three need it handled separately: HTTP/3, request-time DNS lookups with mirror or SSI, and the mail proxy.
- Kubernetes ingress-nginx was retired in March 2026 and will not ship a fix. If your cluster still runs it, turn HTTP/2 off in its configuration now and plan the move.
- NGINX Plus: F5's records list fixed builds per CVE (for CVE-2026-123007, 37.3.0.1 and 37.0.8.1) and no fixed build listed for R33. Read F5's article for your build.
- Not known to be exploited as of 19 November 2026. CISA's exploited list (version 2026.11.18) holds none of the five, and CISA's assessment on 18 November was exploitation none, automatable no, technical impact total. That list records exploitation CISA knows of, so absence is not proof. Xint holds back the final steps of its exploits.
- Block the SCTP kernel module on hosts that do not need it. It is Xint's interim step until a kernel fix exists.

The five flaws, side by side

Affected ranges are nginx.org's (Xint gives 1.13.4 and 0.5.15 as the first releases for the paths it tested in the last two); conditions are F5's CVE records and Xint's write-up. Fixed in 1.31.9 and 1.30.7

CVE	Where	What it needs	Affected	If you cannot patch
CVE-2026-123007	HTTP/2 flow control	HTTP/2 enabled. No account	1.9.5–1.31.8	Turn HTTP/2 off (Xint; F5 lists no workaround)
CVE-2026-123633	HTTP/2, proxying	HTTP/2 from clients, requests proxied to a backend, and the worker running low on worker_connections	1.19.7–1.31.8	Turn HTTP/2 off (Xint; F5 lists no workaround)
CVE-2026-121810	HTTP/3	HTTP/3 (QUIC) listeners, requests proxied to a backend	1.25.0–1.31.8	Turn HTTP/3 off (Xint; F5 lists no workaround)
CVE-2026-123634	Resolver	proxy_pass with a variable in the server name, plus a mirror or SSI subrequest that resolves the same name	0.6.18–1.31.8	Avoid SSI, mirror and variable server names (F5); use fixed upstreams (Xint)
CVE-2026-113425	Mail proxy	A mail block with proxy on and auth_http. No credentials needed	0.5.33–1.31.8	Restrict mail listeners to trusted networks (F5)

All five are memory-safety bugs in the worker process: one out-of-bounds write (CWE-787, from an integer underflow, CWE-191) and four use-after-free (CWE-416). F5's records say each can crash a worker, and four of them that attackers can execute code "on systems with Address Space Layout Randomization (ASLR) disabled or when the attacker can bypass ASLR" (the fifth, CVE-2026-123634, words it a little differently). Xint's exploits bypass it by making nginx disclose its own addresses first.

The same releases fix five more CVEs that Xint did not report, and the stop-gaps below do nothing for them. Only the update closes these.

The other five CVEs fixed in 1.31.9 and 1.30.7 (nginx.org's titles, ratings and ranges)

CVE	What	Rating	Affected
CVE-2026-118899	Infinite loop while reading client request body over TLS 1.3	Major	1.13.0–1.31.8
CVE-2026-120967	SNI bypass when using client certificate validation in stream	Medium	1.25.5–1.31.8
CVE-2026-120993	NULL pointer dereference when using PROXY protocol version 2	Minor	1.31.4–1.31.8 (fixed in 1.31.9)

CVE-2026-125283	Buffer overflow when using alias with try_files	Minor	0.9.6–1.31.8
CVE-2026-123640	Insufficient validation in stream ssl_preread module	Minor	1.15.2–1.31.8

What CVE-2026-123007 actually is

HTTP/2 lets each side tell the other how much data it may send before waiting, a number called the flow-control window. nginx keeps one for the connection and one for each stream. Xint found that when a stream is cancelled, nginx gives the unsent amount back to the connection's window without checking the upper limit. A client that arranges this can push the window past its maximum, and the next size check, done in unsigned arithmetic, wraps around and lets it grow further. nginx then works out a negative length for a chunk of data and passes it to a memory copy as a huge positive one.

A copy that large would normally just crash the worker. Xint shows that on x86-64 with glibc the oversized copy does not simply crash the worker: it overwrites a small area of memory and sends some of it back to the client, which defeats ASLR and gives a controlled write that Xint turns into commands run as the worker. It says the glibc behaviour it relies on has been there since glibc 2.24 in 2016. It does not say what happens on Arm or with musl.

From a worker to root

```
Internet request over HTTP/2 (or HTTP/3, resolver, mail: the other four)
   |  1. nginx2root: CVE-2026-123007 and others (no login)
   v
Commands run as the nginx worker account (nobody, www-data, nginx)
   |  2. Kernel SCTP flaw: no CVE, no patch (Xint, withheld)
   v
Root on the host, and from a container, on the host under it (Xint)
```

Each nginx bug runs code as the worker process, which on most installations is an unprivileged account. That alone is serious. Our reading: the worker sees every request and response passing through it, typically has the TLS keys it serves with loaded in its memory, and can reach whatever the server can reach on your network.

Xint's step to root is a separate flaw in the Linux kernel's SCTP code, present since Linux 4.15 (2018). Xint demonstrated it from inside a Docker container running nginx, ending with root on the host, which means every container on that host. On many distributions the kernel loads the SCTP module the first time any program asks for it, so an unprivileged process can pull it in. There is no CVE or kernel patch for it yet, and Xint is withholding most of the details until there is.

```
# first: is SCTP already loaded? (no output means not)
lsmod | grep -w sctp
# Xint's interim step, for hosts that do not need SCTP
echo 'install sctp /bin/false' | sudo tee /etc/modprobe.d/disable-sctp.conf
```

Who does need SCTP: telecom signalling (SIGTRAN, Diameter, some SIP), some cluster software, and Kubernetes Services declared with protocol SCTP. If lsmod lists sctp, it is already loaded, and the block does not unload it.

This stops the module loading through modprobe. It does nothing if SCTP is already loaded or built into the kernel; check your kernel's configuration (grep CONFIG_IP_SCTP /boot/config-$(uname -r): =m is a module, =y is built in).

How big this is, in numbers

Sources in the last column; the derived figure is labelled

Figure	Value	Source
Hosts Shodan identifies as nginx	56.7 million	Xint, 10 Nov 2026
Of those, reporting a version	11.94 million (21.1%)	Xint
Of the versioned, inside 1.9.5–1.31.8	98.9%	Xint
Versioned hosts in range	About 11.8 million	Our arithmetic on Xint's figures
Years CVE-2026-123007 was in nginx	About 11 (HTTP/2 arrived in 1.9.5, 2015)	nginx.org range; Xint
nginx advisories in 2025	2	Xint, from nginx.org's list
nginx advisories in 2026 to 17 Nov, before these five	26	Xint, from nginx.org's list
CVEs fixed in 1.31.9	10	nginx 1.31.9 release notes
Reporters F5 credits for CVE-2026-123007	4 (Rapid7, Wiz Research, Xint, Ikotas Labs)	The CVE record
EPSS for CVE-2026-123007, 19 Nov	0.064 (percentile 0.94)	FIRST; an estimate, not a measurement

Read the first four rows with care. A version string does not show whether HTTP/2 is on, and a distribution can fix a flaw without changing it: Xint's own example is Ubuntu 22.04, still reporting nginx 1.18.0 at package revision 1.18.0-6ubuntu14.22. The 11.8 million is an upper bound on what a version scan can see, not a count of vulnerable servers. The four credits for one bug say something on their own: more than one team found it, which is a reason not to wait for an exploit to appear.

Who is exposed, and who mostly is not

- Most exposed: nginx that speaks HTTP/2 directly to the internet. A plain server, an API gateway, a load balancer built on nginx.
- Kubernetes clusters on ingress-nginx. Its maintainers retired it in March 2026: "no further releases, no bugfixes, and no updates to resolve any security vulnerabilities". The last controller release is v1.15.1 (19 March 2026) and the repository is archived. Whatever nginx it bundles stays as it is.
- nginx inside a product: appliances, control panels, vendor images. They are fixed by the vendor's release, not by your package manager. Ask the vendor for a date.
- Behind a CDN or load balancer: our reading, not Xint's. If the service in front ends the client's HTTP/2 connection and talks HTTP/1.1 to nginx, the attacker never reaches nginx's HTTP/2 code. If it passes HTTP/2 through, or nginx is also reachable directly, it does. Test the origin itself, not the CDN.
- Distribution packages: Ubuntu 24.04 ships 1.24.0, Ubuntu 22.04 1.18.0, Debian 12 1.22.1 (Xint's list). All are in range for CVE-2026-123007; whether yours is fixed is in your distribution's advisory, by full package version.
- Not affected by CVE-2026-123007: builds without the HTTP/2 module, and 1.9.4 or older (which have other problems).

One thing Xint's write-up does not settle: its exploit for CVE-2026-123007 relies on how glibc copies memory on x86-64. Whether the same path works on Arm or with musl (Alpine images) it does not say. F5's records make no such distinction. Treat every architecture as affected until a source says otherwise.

Things people will get wrong

- Reloading instead of restarting. A reload starts new workers from the old program; only a restart (or a binary upgrade) replaces it.
- Trusting nginx -v. A distribution can carry the fix under the same version number; compare the full package version with its advisory.
- Testing the CDN instead of the origin. The openssl check against the public name only reaches whatever sits in front.
- Turning HTTP/2 off in one server block. Every server block on the same address and port shares the setting; one left on keeps it on.
- Removing quic from the listen line but leaving the Alt-Svc header, so browsers keep trying HTTP/3, or leaving UDP 443 open.
- Treating ingress-nginx as patched by upgrading the chart. There is no fixed release to upgrade to.
- Reading "not on CISA's list" as "safe". It means CISA has not recorded exploitation yet.

Step 1: find out which of the five apply to you

```
nginx -v 2>&1
nginx -V 2>&1 | grep -q -- '--with-http_v2_module' && echo HTTP2_MODULE=present || echo
HTTP2_MODULE=absent
sudo nginx -T 2>&1 | sed 's/#.*//' | grep -nE 'listen[[:space:]].*
(http2|quic)|^[[:space:]]*http[23]
[[:space:]]+on|mirror|resolver|ssi[[:space:]]+on|proxy_pass[[:space:]]+
[^;]*\$|^[[:space:]]*mail[[:space:]]*\{|auth_http'
# full package version, to compare with your distribution's advisory
dpkg -l 'nginx*' 2>/dev/null | grep ^ii; rpm -qa 'nginx*' 2>/dev/null
```

From a machine outside your network, check whether the nginx port itself accepts HTTP/2 (Xint's test; replace HOST):

```
openssl s_client -connect HOST:443 -servername HOST -alpn h2 </dev/null 2>&1 | grep -E 'ALPN
protocol:|No ALPN negotiated'
```

ALPN protocol: h2 means HTTP/2 is reachable. If a CDN sits in front, run it against the origin's own address too.

Step 2: patch, then restart

Install 1.31.9 or 1.30.7, or your distribution's fixed package, and restart nginx through your service manager. A reload starts new workers with the old program, and connections already open stay with the old workers: in ps they show as "nginx: worker process is shutting down" until they close. After the restart, nginx -v and the package version should both show the fix. A restart drops every connection in flight, so pick the moment for busy servers; Xint names a live binary upgrade as the alternative. Rebuild container images from a fixed base image and redeploy, then check the running container.

Step 3: if you cannot patch today

Stop-gaps per flaw, with what each costs. They block the paths Xint knows about; they are not a substitute for the update

You use	Change	Blocks	Cost
HTTP/2	On 1.25.1 and later: http2 off; in each server block AND remove the http2 parameter from its listen lines. Change every server block sharing the same address and port. Older: remove http2 from the listen lines	CVE-2026-123007, CVE-2026-123633	Browsers fall back to HTTP/1.1; native gRPC stops working (Xint)

HTTP/3	Remove the listen lines with quic, set http3 off;, and replace any Alt-Svc header with add_header Alt-Svc 'clear' always;. Then sudo ss -lunp 'sport = :443' must not show nginx; Xint also suggests blocking UDP 443 at the firewall	CVE-2026-121810	Clients use HTTP/2 or HTTP/1.1 over TCP instead
proxy_pass with variables plus mirror or SSI	Fixed upstream names; no request-time DNS. Turning mirror off alone is not enough (Xint)	CVE-2026-123634	Configuration work; dynamic routing by host name stops
Mail proxy with auth_http	Bind the mail listeners to loopback or allow only trusted addresses in your firewall	CVE-2026-113425	Outside mail clients cannot connect

After any of these, restart rather than reload, for the same reason as in Step 2, then re-run the openssl check from Step 1: it should print No ALPN negotiated. Empty output can also mean the connection failed, so load the site to be sure. These changes cover only Xint's five; the other five CVEs in the same release, including CVE-2026-118899 (rated major), close only with the update. Turning HTTP/2 off is a stopgap for days, not a setting to keep.

Step 4: if you run ingress-nginx

Our recommendation, since the project can no longer give one. Turn HTTP/2 off in the controller's ConfigMap now, which closes CVE-2026-123007 and CVE-2026-123633 at the cost of HTTP/2 to clients. The key is use-http2; confirm it against the documentation for your controller version before you apply it.

```
# installed with Helm: set it in values, or the next helm upgrade reverts a manual patch.
# Pin the chart you already run, so this does not also upgrade the controller (release name
and version: helm list -n ingress-nginx)
helm upgrade ingress-nginx ingress-nginx/ingress-nginx -n ingress-nginx --version
YOUR_CHART_VERSION --reuse-values --set-string controller.config.use-http2=false
# installed from manifests: patch the ConfigMap
kubectl -n ingress-nginx patch configmap ingress-nginx-controller --type merge -p '{"data":
{"use-http2":"false"}}'
# then restart the controller (a Deployment or a DaemonSet) so open connections close
kubectl -n ingress-nginx rollout restart deployment ingress-nginx-controller
# or, for a DaemonSet: kubectl -n ingress-nginx rollout restart daemonset ingress-nginx-
controller
```

The restart drops connections in flight, as any nginx restart does. Then plan the move. The Kubernetes maintainers recommend Gateway API, "the modern replacement for Ingress". If you move to another controller built on nginx, check that it has shipped a fixed nginx before you rely on it. A cluster that stays on a retired controller will meet the next nginx flaw with no fix either.

Step 5: check whether it already happened

From Xint's guidance. The frames that trigger CVE-2026-123007 are not requests, so the access log shows almost nothing. Look elsewhere:

- Connections: one outside address holding a TLS connection for minutes while bytes received grow fast and bytes sent stay small, alongside many short connections from the same address (sudo ss -tinp state established '(sport = :443)', twice, a short interval apart).
- Requests per connection: Xint gives a log format that records the connection, protocol, request length and bytes sent, so the summed request length per connection can be compared with what ss shows the connection received. A large gap is the frames the access log does not show.
- Worker crashes: a failed attempt can kill a worker (sudo grep -Ei 'worker process.*exited|signal 11|segfault' /var/log/nginx/error.log; sudo coredumpctl list nginx). A worker can crash for other reasons too.
- Programs started by workers: a worker has no reason to start a shell, and Xint calls it the clearest sign of a successful attack. With auditd, record execve by the worker's user id and review it (Xint gives the rule). An empty result covers only the time after you add the rule.

If you find something: keep the logs and, if you can, a disk image before rebuilding. Treat the TLS keys the server held as exposed and reissue the certificates, rotate the credentials the server could reach, and, on a host where SCTP could load, treat the host and its other containers as possibly compromised to root.

What to send your vendors today

> *We run [product, version], which includes nginx. Which nginx version does it ship, and does it include the fixes in nginx 1.31.9 or 1.30.7 of 18 November 2026 for CVE-2026-123007, CVE-2026-123633, CVE-2026-121810, CVE-2026-123634 and CVE-2026-113425? If not, which release will, and when? Until then, can HTTP/2 and HTTP/3 be turned off on it, and how?*

Check every server and every project with lattice-scan

Steps 1 to 5 are done by hand, one server at a time. To do the same over a fleet, and to keep a record, use the open-source lattice-scan command line. Run on the server itself, it reads the installed packages, snaps, the hardware, the running container images with their operating system and packages, and the kernel, compares them with the CVE ids you name, and sends only those ids out. Naming the CVEs needs no key. These five are in our recent-CVE list through 24 November, so lattice-

scan recent shows them too, including nginx inside running container images, which is where it often hides.

```
# on each nginx server, after patching and restarting
lattice-scan --cve CVE-2026-123007 --cve CVE-2026-123633 --cve CVE-2026-121810 --cve CVE-2026-123634 --cve CVE-2026-113425
# or: everything recent, exploited or critical
lattice-scan recent
```

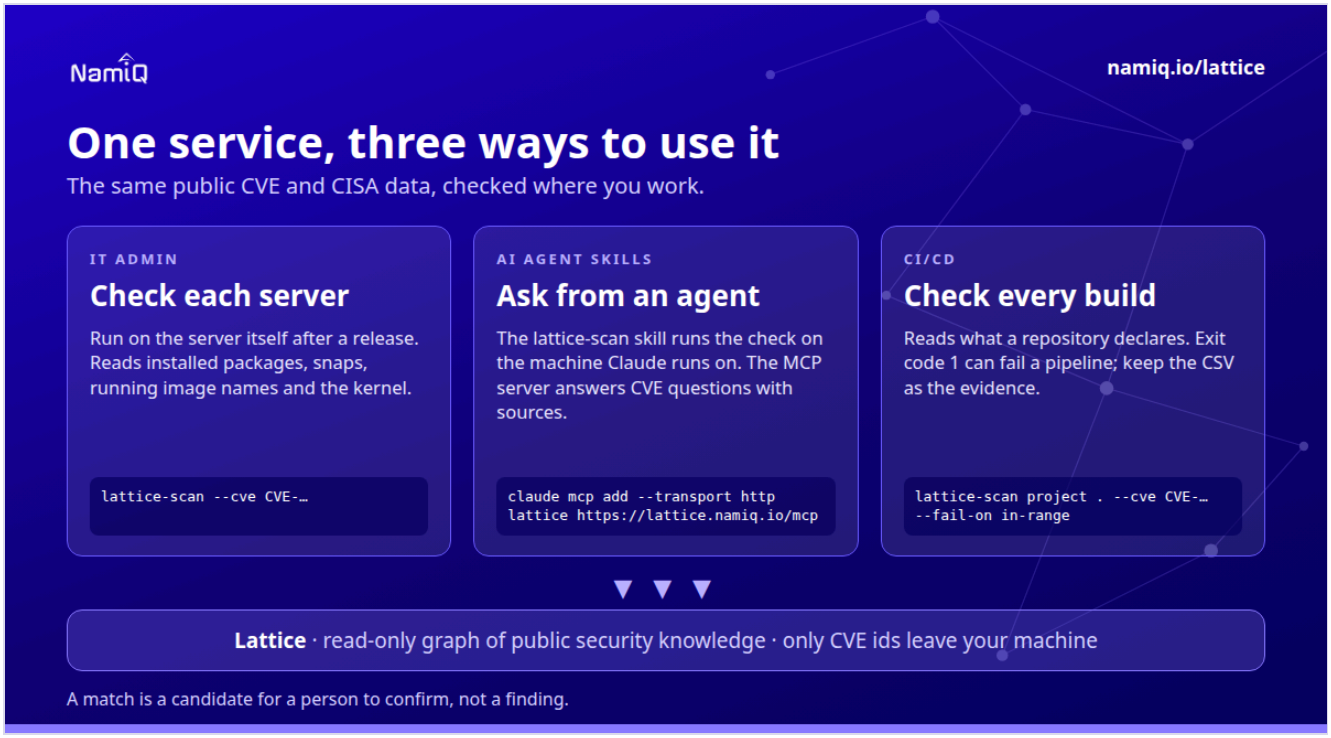
A match is a candidate for a person to read, not a finding: it says nginx is installed, not whether HTTP/2 is on, so Step 1 is still yours. A distribution can fix these under an unchanged version number, the same trap as in Step 1, so on Debian-family hosts read the changelog line the report shows. Naming one of these ids returns no version check yet, because NVD's record for them has not reached the graph. One thing that does not flatter us: our recent list left these five out of its first runs after publication, because it kept only its 40 highest-scored new records that day and the nginx CVEs were not among them. That was corrected on 19 November.

For the code you ship, the project check reads Dockerfiles and compose files, such as a FROM nginx:1.27 line. A tag without a patch number cannot show whether the fix is in, so confirm the image that is actually running.

```
lattice-scan project . --cve CVE-2026-123007,CVE-2026-123633,CVE-2026-121810,CVE-2026-123634,CVE-2026-113425
```

Who can use it, and how

For	What to do	Command
IT admin	Run the host check on each server after a release, and again when a new entry lands.	lattice-scan --cve CVE-2026-123007 --cve CVE-2026-123633 --cve CVE-2026-121810 --cve CVE-2026-123634 --cve CVE-2026-113425
AI agent skills	The lattice-scan skill runs the check on the machine Claude runs on. The Lattice MCP server answers questions about the CVEs with their sources.	claude mcp add --transport http lattice https://lattice.namiq.io/mcp
CI/CD	Check the repository on every pull request. Exit code 1 fails the pipeline; keep the CSV as the evidence.	lattice-scan project . --cve CVE-2026-123007,CVE-2026-123633,CVE-2026-121810,CVE-2026-123634,CVE-2026-113425 --fail-on in-range



Where the same check is used. The three ways read the same public data; only CVE ids leave your machine.

What we do not know yet

- The kernel SCTP flaw: no CVE id, no patch and no list of fixed kernels yet.
- Whether anyone is using these. CISA's assessment on 18 November was exploitation none; that can change without notice, and four credited reporters for CVE-2026-123007 suggest others can find it too.
- How many of the 56.7 million hosts have HTTP/2 enabled and reachable. The version figures cannot say.
- Exploitability on Arm and with musl. Xint's demonstrated path uses x86-64 glibc behaviour; F5 does not distinguish.
- When products that embed nginx ship fixes. Each vendor's own advisory decides it.

If you take one thing from this: the update is one package and one restart. The work is the nginx you did not know you ran, inside images, controllers and appliances, and the one that cannot be fixed at all because its project has stopped. If that list takes more than an afternoon to produce, fix that too.

Appendix 1. Severity decoded

CVE-2026-123007 (F5; the same vectors on all five records): score 9.2, `CVSS:4.0/AV:N/AC:H/AT:N/PR:N/UI:N/VC:H/VI:H/VA:H/SC:N/SI:N/SA:N` Network, no login, no user action, high complexity: the attacker has to make nginx disclose its own addresses first. Impact stays on the nginx host in the vector; the kernel step to root is a separate flaw. CVSS 3.1: 8.1, CVSS:3.1/AV:N/AC:H/PR:N/UI:N/S:U/C:H/I:H/A:H.

Metric	Value	What it means here
Attack vector	Network	Reachable over a network, with no need to be on the same segment.
Attack complexity	High	Needs conditions outside the attacker's control.
Attack requirements	None	The scorer recorded no deployment condition; this note lists the set-up each flaw needs (HTTP/2 on, for this one).
Privileges required	None	No account is needed.
User interaction	None	No one has to do anything.
Vulnerable system: confidentiality	High	Total loss of confidentiality.
Vulnerable system: integrity	High	Total loss of integrity.
Vulnerable system: availability	High	Total loss of availability.
Subsequent systems: confidentiality	None	The score counts no impact beyond the nginx host; the kernel step in this note is outside it.
Subsequent systems: integrity	None	As above: the kernel step to root is outside the score.
Subsequent systems: availability	None	As above.

Appendix 2. Timeline

Date	Event
11 October 2026	Xint's researchers find the flaws: CVE-2026-123007 first, while preparing for ZeroDay Cloud 2026, the other four a few days later with an automated scan (Xint).

16 October 2026	Reported to F5 through HackerOne; validation completed on 20 October (Xint).
27 October 2026	Reports delivered to F5's nginx development team (Xint).
9 to 10 November 2026	The nginx team asks Xint to validate the patches; Xint completes it on 10 November. CVE identifiers reserved 9 November (CVE records). Xint's Shodan count is taken on 10 November.
18 November 2026	nginx 1.31.9 (15:54 UTC) and 1.30.7 (15:56 UTC) are released with fixes for ten CVEs. F5 publishes the five CVE records (15:23 UTC) and Xint its write-up. CISA's SSVC entries: exploitation none, automatable no, technical impact total.
19 November 2026	None of the five is in CISA's catalogue (version 2026.11.18). FIRST's EPSS gives CVE-2026-123007 0.064 (percentile 0.94). The kernel SCTP flaw has no CVE and no patch.

Appendix 3. Exposure by situation

These priorities are our judgement from the attacker prerequisites in the sources. They are not vendor ratings.

Your situation	Priority	Timeframe
nginx speaks HTTP/2 directly to the internet (a server, gateway or load balancer)	Critical	Update and restart today; if you cannot, turn HTTP/2 off today
Kubernetes on ingress-nginx	Critical	Turn HTTP/2 off in the controller's ConfigMap today; plan the move off a retired controller this quarter
HTTP/3 (QUIC) listeners proxying to a backend	High	Update this week; until then remove the quic listeners
Behind a CDN or load balancer that ends HTTP/2 and talks HTTP/1.1 to nginx, origin not reachable directly	Medium	Update this week; test the origin's own address to prove it is not reachable
proxy_pass with a variable plus mirror or SSI, or the mail proxy with auth_http	High	Update this week; until then use fixed upstreams, or limit the mail ports to trusted networks
nginx inside a vendor product or appliance	High	Ask the vendor for the fixed release and a date (message below)
Internal only, HTTP/2 on	High	Update this week; one compromised machine on that network is enough

Internal only, HTTP/2 and HTTP/3 off, no mail, resolver, mirror or SSI	Low	Update at the next maintenance window; 1.31.9 also fixes five other CVEs

Appendix 4. Definition of done

- Every nginx is listed, including those inside container images, ingress controllers and vendor products, with the full package version and the date.
- For each, the configuration has been read for http2, quic and http3, the resolver with mirror or SSI, and mail with auth_http, and the result is written down.
- Every nginx is on 1.31.9, 1.30.7 or a distribution package whose advisory names all ten CVEs fixed in 1.31.9, read from the running binary, and has been restarted (not reloaded) since.
- Where an update must wait, the stop-gap for each flaw that applies is in place and the openssl check from outside shows no h2 on the origin itself.
- ingress-nginx clusters have HTTP/2 off, confirmed on a live connection, and a dated plan to move off the retired controller.
- Hosts that do not need SCTP have the module blocked, and lsmod showed it was not already loaded.
- Worker crashes and long-lived connections since 18 November have been looked at on internet-facing servers, and an execve audit rule is in place on the most exposed ones.
- A date is set to re-read Xint's page and the kernel mailing lists for the SCTP fix.

Sources

- Xint: nginx2root, remote code execution (18 Nov 2026): https://xint.io/blog/nginx2root-remote-code-execution
- nginx security advisories: https://nginx.org/en/security_advisories.html
- nginx 1.31.9 release notes: https://github.com/nginx/nginx/releases/tag/release-1.31.9
- F5 K000163926 (CVE-2026-123007): https://my.f5.com/manage/s/article/K000163926
- F5 K000163928 (CVE-2026-123633): https://my.f5.com/manage/s/article/K000163928
- F5 K000163927 (CVE-2026-121810): https://my.f5.com/manage/s/article/K000163927
- F5 K000163925 (CVE-2026-123634): https://my.f5.com/manage/s/article/K000163925
- F5 K000163931 (CVE-2026-113425): https://my.f5.com/manage/s/article/K000163931
- CVE records for the five CVEs (F5; CISA SSVC entries): https://www.cve.org/CVERecord?id=CVE-2026-123007
- CISA Known Exploited Vulnerabilities catalog (version 2026.11.18: none listed): https://www.cisa.gov/known-exploited-vulnerabilities-catalog
- FIRST EPSS (scores read 19 Nov 2026): https://www.first.org/epss/
- Kubernetes: Ingress NGINX retirement (11 Nov 2025): https://kubernetes.io/blog/2025/11/11/ingress-nginx-retirement/
- kubernetes/ingress-nginx (archived): https://github.com/kubernetes/ingress-nginx