ultraviolet

THREAT ADVISORY SPECIAL REPORT

AtSign SSHNPD Vulnerabilities

Services Performed By:
UltraViolet Cyber TIDE Team
tide@uvcyber.com

Published Date:
August 25, 2026
TLP:GREEN

Executive Snapshot

Atsign is the company behind atProtocol and NoPorts, a remote-access platform built around the idea of removing exposed inbound ports as the primary trust boundary. Rather than granting access based on network reachability, such as whether a device can be reached on a given IP and port, NoPorts verifies cryptographic identity and establishes end-to-end encrypted sessions through relay infrastructure. The Atsign Foundation, a nonprofit, maintains the open-source protocol specification and reference implementations, including the sshnpd daemon at the center of this disclosure, making the codebase directly relevant to the security guarantees the product advertises.

Aaron/”Zonifer,” an UltraViolet Cyber Offensive Security Services Team member, discovered two vulnerabilities in Atsign's NoPorts C daemon (sshnpd) that, when chained, allowed an attacker holding any valid atSign identity, not just an authorized manager, to inject a malicious SSH key into a target's authorized_keys file and reach that system's SSH service through NoPorts' own relay, despite the service never being exposed to the network. The root cause was a missing authorization check in the request-dispatch path combined with an inverted validation function that caused SSH public-key checks to fail open. Atsign patched the issue within four days of disclosure, and a CVE is pending, but any organization running the C implementation should treat this as a priority action item. UVCyber Threat Intelligence and Detection Engineering (TIDE) Team have begun implementing Detections for this vulnerability and performing Threat Hunts surrounding AtSign infrastructure in UVCyber customers environments.

- Patch immediately: Update all NoPorts C-daemon deployments to the fixed release and confirm the version in use no longer matches the vulnerable build before assuming remediation is complete.
- Audit authorized_keys and manager configuration: Review authorized_keys files on systems that ran the affected daemon (especially with -s enabled) for unrecognized keys, and verify the --manager list reflects only intended, trusted atSigns.
- Add negative authorization testing: For NoPorts and any similar identity-first remote-access tooling, build tests that confirm authenticated-but-unauthorized identities are explicitly denied, not just that authorized identities succeed, particularly when logic is ported across languages or implementations.

The original vulnerability disclosure article by Aaron/”Zonifer” can be found at:
https://zonifer.dev/posts/noports.html

TIDE Team Analysis

UltraViolet Cyber security researcher Aaron/”Zonifer” disclosed two critical implementation flaws in the C daemon (sshnpd) of Atsign's NoPorts, a remote-access platform that replaces exposed SSH listeners with an identity-first, cryptographically authenticated connection model. The findings, published August 25, 2026, show that authentication and authorization were not enforced together in the C implementation, allowing an authenticated but unauthorized identity to reach functionality reserved for trusted administrators. Atsign responded quickly: the report was filed August 21, a patched release shipped August 25, and a CVE request is pending.

The first flaw is a missing authorization gate in the daemon's request-dispatch logic. The C daemon verifies that a message was cryptographically signed by a legitimate atSign, but it never checks that atSign against the configured manager list before executing session-request, SSH-request, and tunnel-request handlers. The project's Dart implementation contains this exact check; the C port simply left it out. The result is that any holder of a valid atSign identity, not just an authorized manager, can request relay connectivity to a service that was never exposed to the network in the first place.

The second flaw compounds the first when the daemon runs with the -s flag, which enables SSH public-key sharing. The validation routine intended to confirm that a submitted key begins with a supported SSH key prefix contains a chain of errors: it measures the wrong string, applies a bounds check that can never trigger, and inverts the result of strncmp() so that mismatches are treated as valid matches. The practical effect is that the prefix check fails open, allowing arbitrary attacker-supplied key material to pass through to the code that writes into authorized_keys.

Chained together, these two bugs allow an attacker with any valid atSign to request a session with an unauthorized identity, submit a malicious SSH public key that the broken validation accepts, have that key written into the target's authorized_keys, and then use the daemon's own relay to reach localhost:22 and authenticate with the key they just installed. No memory corruption, credential theft, or protocol manipulation is required. The attacker is using the product's intended control flow exactly as designed, which is what makes the chain both simple to execute and difficult to detect through conventional means.

For enterprise environments, this case illustrates a structural point rather than a one-off bug: removing an exposed listening port does not remove the need for authorization, it relocates that decision into the application's control plane. Organizations running NoPorts to reach systems with no inbound SSH access, no VPN termination, and no exposed management surface were nonetheless one authorization gap away from unauthorized access to those same systems. Traditional network scanning would not have surfaced this exposure, since the attack surface lived in message-handling logic rather than an open port. Any environment using the affected C daemon, particularly with -s enabled for public-key sharing, should treat this as a priority patch and review whether unauthorized SSH key material was added to authorized_keys during the exposure window.

The broader lesson for CTOs and CISOs evaluating identity-first or "zero exposed port" architectures is that cryptographic authentication and authorization are separate controls that must both be enforced at every sensitive boundary, and that porting security logic between languages or implementations (here, Dart to C) creates real risk if enforcement doesn't carry over identically. Vendor response in this instance was strong, with a four-day turnaround from report to patched release, but the incident is a useful prompt for internal teams to confirm patch adoption, audit manager-list configurations, and build negative authorization testing (confirming that unauthorized identities are correctly denied, not just that authorized ones succeed) into their own security-sensitive remote-access tooling.

Why It Matters

NoPorts is typically deployed in environments where administrators need remote access to infrastructure that is intentionally kept off the public internet, servers sitting behind restrictive data center firewalls with no inbound SSH listener, no VPN termination, and no exposed management services. Development teams and enterprises adopt this model specifically to shrink their network attack surface: instead of opening a port and relying on firewall rules or VPN gateways to gate access, the device only ever makes outbound connections, and the NoPorts daemon decides who is allowed to request a session based on cryptographic identity. This is attractive for production systems, CI/CD infrastructure, internal tooling, and any host where exposing SSH directly, even behind a VPN, is considered an unacceptable risk. The tradeoff is that the daemon itself becomes the access-control boundary, so the security of the entire architecture depends on that daemon correctly enforcing who is authorized, not just who is authenticated.

These vulnerabilities undermine exactly that guarantee. Because the C daemon authenticated requests without checking them against the configured manager list, the core security promise of NoPorts, that only trusted identities can reach the protected service, did not hold in practice. Any organization relying on the C implementation to protect systems with no other network-level defenses was exposed to unauthorized relay access and, where public-key sharing was enabled, potential SSH credential injection, regardless of how well the surrounding network was locked down. For enterprises that adopted NoPorts as a substitute for traditional perimeter controls rather than a complement to them, this means the systems most dependent on NoPorts for protection, precisely because they had no other exposed access path, were also the most exposed to this flaw. This case is a useful reminder that identity-first architectures move risk into application logic rather than removing it, underscoring why rigorous authorization enforcement is just as critical to these designs as the cryptographic identity model itself.

How to Respond

- Strictly adhere to cybersecurity fundamentals and ensure all personnel undergo annual phishing and social engineering training. Speak with your UltraViolet Cyber TAM Representative to schedule a live phishing engagement.
- Update all NoPorts C-daemon deployments to the fixed release and confirm the version in use no longer matches the vulnerable build before assuming remediation is complete.
- Perform annual tech refresh reviews to gain a holistic understanding of your infrastructure. Speak with your UltraViolet Cyber TAM Representative to schedule a Red Team or Purple Team engagement to gain insight into the vulnerabilities in your environment.

What UltraViolet Cyber is Doing

- Tracking new CVEs and high impact vulnerabilities, analyzing and deploying public Proof-Of-Concept code against custom built targets.
- Proactively enabling custom detections based on the collected artifacts, tactics, techniques, and procedures identified in this activity. Given that this vulnerability was discovered by a UVCyber researcher, these detections are unique and exclusive to UVCyber Managed SOC customers.
- Performing hypothesis driven threat hunts based on threat actor behavior and artifacts. UVCyber customers will be informed of the results through secure channels.
- Parsing available victim dump data for any social, financial, business, or technical relations to UVCyber Clients and partner organizations.
- Aggregating threat intelligence from myriad sources and applying the most up-to-date knowledge to proactive threat hunting and response.

About UltraViolet Cyber

UltraViolet Cyber is a leading tech-enabled managed security services provider, delivering unparalleled cybersecurity expertise that fills technology and talent gaps across Global 2000 and Federal Government customers. Founded and operated by security practitioners from the national intelligence community, UltraViolet Cyber connects offensive security, application security, detection and response, and security engineering to deliver a differentiated approach to cybersecurity operations. Transforming customers' security programs, UltraViolet Cyber's flagship security-as-a-service solution, UV Lens, removes complex operational silos, replacing them with integrated security capabilities. UltraViolet is headquartered in McLean, Virginia with technology centers across the world.

443.351.7630 / info@uvcyber.com | UltraViolet Cyber | @uv_cyber

---