COMMUNITY CONTENT

Submit #782904: AntaresMugisho PyBlade v0.1.8-alpha through v0.2.0-alph Code Injection

Title	AntaresMugisho PyBlade v0.1.8-alpha through v0.2.0-alph Code Injection
Description	This code is vulnerable to CWE-94: Code Injection and CWE-1336: Template Engine Injection due to unsafe expression evaluation in template rendering. The vulnerability affects v0.1.8-alpha through v0.2.0-alpha through two different mechanisms: v0.1.8-alpha and v.0.1.9-alpha: The _is_safe_ast() function in sandbox.py contains a logic flaw. The attribute whitelist check only validates ast.Name nodes (e.g., str.method) but bypasses ast.Constant nodes (e.g., ''.__class__), allowing access to dangerous Python magic methods. v0.2.0-alpha: The evaluator.py file uses eval() directly without any AST validation, providing no security checks at all. This allows an attacker to achieve Remote Code Execution (RCE) through Python's object model by accessing __class__, __mro__, and __subclasses__ chains.
Source	https://github.com/AntaresMugisho/PyBlade/issues/1
User	zhangxinyu06 (UID 90407)
Submission	03/19/2026 10:42 AM (17 days ago)
Moderation	04/04/2026 03:54 PM (16 days later)
Status	Published
VulDB entry	[AntaresMugisho PyBlade 0.1.8-alpha/0.1.9-alpha AST Validation sandbox.py _is_safe_ast special elements used in a template engine]
Points	20

Community Content

Submissions are made by VulDB community users. VulDB is *not responsible* for their content nor the links to external sources.

Please use the raw information shown and the links listed *with caution*. They might contain malicious and harmful actions, code or data.

The corresponding VulDB entries contain the moderated, verified, and normalized information provided within the raw submission.

Documentation

- Submission Policy
- Data Processing
- CVE Handling