unblu

Unblu security bulletins

Unblu takes security very seriously. If a vulnerability is brought to our attention, we aim to fix it as soon as possible.

Known vulnerabilities are published as security bulletins on this page. However, we actively notify our customers before we publish vulnerabilities. This allows them to take appropriate steps to protect their own customers.

List of security bulletins

- UBL-2026-004: Cross-site scripting in embedded co-browsing
- UBL-2026-002: Authorization bypass in configuration cascade operations
- UBL-2026-001: Open redirect and DOM-based XSS via the `redirectOnFailure` parameter
- UBL-2025-002: File upload functionality possible even when disabled
- UBL-2025-001: Replace uploaded files knowing the file upload ID

Reporting vulnerabilities

If you discover a vulnerability or any other security issue in any Unblu products, please report it to security@unblu.com. You can use the PGP key below to encrypt your email.

Unblu runs a bug bounty program on YesWeHack. If you're interested in taking part, request an invitation from security@unblu.com. Make sure to mention your YesWeHack username in your message.

Listing 1. PGP key for reporting problems to security@unblu.com

TXT

```
-----BEGIN PGP PUBLIC KEY BLOCK-----

mQGNBGfz0rgBDADSjfl7UPwT2N4cuptdW1uV4Lg7z+G5kJAM9K8qUHNnOyAgmGv7
6YkvzqKQta+JLRC/0NVdM0/2sMEl6eeooxPb8go+MTdrEpoEGtXhiitroqOKyWXz
pi1I2oVWcnf8JHN4VzkLKV7MlXqBxFajzRIdW10JpTkNJckG99C0XG7bhP6gKG++
WQ1ypQYD7lE7obRbXJQgjlNPaRn2bS5y6byWL9ZgKR7P6IDjL6jLPEofXDpXs7Wd
E12+wN/Ru5EPybxlB5Rb4H8gk8qnZX0COpyZKsZAig6mc1K+O7xa2eDHpkQ7//P8
dVN8UFsLzPhYG6rfJ2aNH6q166CTNxx8O9RoulK8k404RZEYEdrCgTXKjqwsq9Ue
Qz5YEIQP0ajVoJ6bLiMD6okMmPV0huv+fRQ1TryCeqkkubdbmv9dtmAgXZGC92F2
BEyLn2ksRKYMoYUYXDHtfGNvklbtqVr4E9UcsP+d76iJPA2j0EwgO6p9kL56QWT5
Kgc8GVf4Ci0Y54cAEQEAAbQjVW5ibHUgU2VjdXJpdHkgPHNlY3VyaXR5QHVuYmx1
LmNvbT6JAdQEEwEKAD4WIQQ+thE6qJ1o0pMdz2f236cjI7OsMwUCZ/PSuAIbAwUJ
A7U4AAULCQgHAgYVCgkICwIEFgIDAQIeAQIXgAAKCRD236cjI7OsMxueC/94VBv3
IkM+uKjtrsL6tRbmCXuSv2M14N3saMy1KiifXQ60XvJpy6Jdgj8sYouDKXQIlbCv
jcUXe8iHbUT95Rwe549UVBcExTmkFzrXtHmH3MUU8L751kHN2S/XXcgxnxd5jWPD
lKs7295Hpd12xHNUXXCgLbgVYcFFpLMYzBQDoOXb8l50gtaQFIFf47BbSs7RdPkr
8XhBF1RRF9p05DqBeUzlvmbmtXdpG0kpIT20NmwUM5NOhDZdH62kq6MYES43X3HS
iS/9/AEXmegYRwQm4ZXpCmkR/GYD1EStn77EAFl3pK6wyHxMyR9o2JiqbYyBZTbK
eq+R/akKWnrxRRwb0rVTox5+nkHEPAiTrZ80qlR437mWx+aQycTOp70gg9pOKC4T
Uu+ZD0wx29D8j3+bABjz98yb7Q0GBfqAqtSRCH02Di4yyy6/+2nTD4rIuoJutQUP
D377sMm8OTzDqu9htYxX37AAF0kT1EXXeyrFiT94jzBz9tSAzyqoqqeLcr+5AY0E
Z/PSuAEMALgf2klcYWqlatN99cQEuioh09d67XQiUItKHDln8xvZt0bVmV/bSQg+
S1SgFlr75oX0W8WvSsD0UZOGWKOaGpsO0zev8Ibtn/esKaLQJmmaiLhmo0qv9V0I
mpV8NK7azHkSUFfS5emIYn0WZYai/NKIvkzjTD+YB616upL6hkkLVUnDSfXWHGCC
LrX9tQKtcLjPofiq1k9qJOvXZMfuFLohVBvAsVBpP57eCmBvAQX6WbaD9sKnkcwJ
b5GU4JpVAJzlRY/XNh8H0DZP7SD4Pwc8WPATdFY6LpDHGh41VYYchunX3b27G8IQ
l2+/B7sCXWfRx9qOTSTRpwW9orsmFNKMPpfqtXrrUwenJFWuOewYAekjxNQQiYM7
6qm5yMThZPqaIMBPy2gnG5Vxbwlv73DJWUcl+e1S60g236rdBefyiCumJNs9e2Hf
+170hre6iFcH8teU2FmMLtGj3OEpFlxjPIPvc+QxkgYo/TkMMgRypqXAKtgay1rj
2gWD93mqIwARAQABiQG8BBgBCgAmFiEEPrYROqidaNKTHc9n9t+nIyOzrDMFAmfz
```

```
0rgCGwwFCQO1OAAACgkQ9t+nIyOzrDMkNwv/ccA89Kf6qO/Ki/BgzFTDgwX+5Im+
/7HElrLoG4deSn/KKeQyAAh/TFpE4d8UQHfgbAfxlAyPyl66PhtxGZL+XBCBtZdF
HT1lEayuFx4DqDYU36jcRUq0oLpuMZSolgnm5sW8JLCyijM6Xu225cc3j0UEzRr0
rl5oZneSHwVYl5a20X6mr6Grwzjh/4OCwdMMecX3nr1AlWAQXrEATsWnK6VcxstU
w0yAek7CHzmTM6E3VWtWypL8niML+HtAZboymuSQbQEG2CuP8qP69GfdKGwim0PH
F4+40ZIJXyLoXPp1sF4WNXSWoRWlkQ6EbcLdOrgHYxyMVwHvx9DY0OV99IuJSD4j
UOyIzy2dS3i5Lg/RaU0b4S5Iiyr4Pr0Y1RzwkXKaLE2o/+rqG9/xC8jTs3D2rzRe
9lnThUphF/dfs5UtTi2IcPtyon04t9aB0G6UhBfQdOQu8xS66LT2BLdam+jel+jS
RcxH52nPKNzsZaJOecWnG+BwMhmxYGe6jebw
=Eogv
-----END PGP PUBLIC KEY BLOCK-----
```

Security bulletins

UBL-2026-004: Cross-site scripting in embedded co-browsing

Published on: 2026-10-15

CVE ID: CVE-2026-14933

Severity: 8.6 High

Credits: Kenan Karalioglu (chef_shell), via YesWeHack

Summary

An attacker who alters a page during an embedded co-browsing session can run JavaScript in the agent’s browser.

Affected components and versions

- Unblu Spark (v8): 8.39.2 and all earlier versions, except the versions listed under Fix versions below

- Unblu Spark (v7): 7.57.7 and all earlier versions

Fix versions

- Unblu Spark (v8): 8.35.2-hotfix.0, 8.37.2-hotfix.2, 8.38.2-hotfix.0, and 8.39.3
- Unblu Spark (v7): 7.57.9

Details

During embedded co-browsing, the Collaboration Server removes executable code from the Document Object Model (DOM) of the visitor's page before it transmits the DOM to the agent's browser. However, the agent's browser renders the selected option of dropdown lists (`<select>` elements) on the co-browsed page as HTML rather than as plain text. An attacker in the visitor role can alter the page, for example with the browser's developer tools, to make the agent's browser run JavaScript the attacker supplies. The JavaScript runs in the context and under the identity of the agent's session in the Agent Desk or the Single Conversation Desk.

Mobile co-apping isn't affected.

In the fix versions, the agent's browser renders the selected option as plain text.

Workarounds

Unblu Cloud customers needn't take any action. The fix is already deployed in Unblu Cloud.

If you can't upgrade an on-premises deployment immediately, disable embedded co-browsing by setting com.unblu.messenger.embeddedCoBrowsingEnabled to `false`.

UBL-2026-002: Authorization bypass in configuration cascade operations

Published on: 2026-10-15

CVE ID: CVE-2026-9589

Severity (on-premises): 8.9 High

Severity (Unblu Cloud): 9.3 Critical

Credits: Kenan Karalioglu (chef_shell), via YesWeHack

Summary

A registered user can change configuration properties they aren't allowed to change, including properties that belong to other users or other accounts.

Affected components and versions

- Unblu Spark: 8.36.1-hotfix.0 and all earlier versions

Fix versions

- Unblu Spark: 8.37.2 and all later versions

Details

Unblu reads and writes configuration properties along the configuration cascade. The Collaboration Server doesn't check whether the user reading or changing the configuration has access to the entity it belongs to. As a result, the following attacks are possible:

- An admin can change the configuration of another account.
- A user can change the configuration of another user, for example, that user's personal preferences.
- With certain nondefault configuration validation settings, a registered user can write configuration properties they aren't allowed to edit, in any scope.

In Unblu Cloud, the other account can belong to a different customer. On-premises installations rarely have more than one account, but those that do are affected in the same way.

In the fix versions, the Collaboration Server checks whether the user has access to the relevant entity on every configuration read and write. It also rejects writes to undefined configuration properties, and writes of values that violate a property's restrictions, regardless of the configuration validation settings.

Workarounds

Unblu Cloud customers needn't take any action. The fix is already deployed in Unblu Cloud.

There's no workaround for on-premises deployments. Upgrade to a fix version as soon as possible.

UBL-2026-001: Open redirect and DOM-based XSS via the `redirectOnFailure` parameter

Published on: 2026-07-22

CVE ID: CVE-2026-8152

Severity (on-premises, same-origin deployments): 9.3 Critical

Severity (Unblu Cloud): 7.0 High

Credits: Kenan Karalioglu (chef_shell), via YesWeHack

Summary

Unblu Spark contains an open redirect vulnerability that an attacker can escalate to a DOM-based cross-site scripting (XSS) attack.

Affected components and versions

- Unblu Spark: 8.36.1 and all earlier versions

Fix versions

- Unblu Spark: 8.36.1-hotfix.0 and all later versions

Details

The `redirectOnFailure` URL parameter isn't validated against an allowed URL scheme, and it isn't passed through the redirect filter that applies to other user-supplied redirect targets. As a result, an attacker can craft a link that redirects a victim's browser to an untrusted destination or runs arbitrary JavaScript in the victim's browser.

When you deploy Unblu Spark with `com.unblu.identifier.siteEmbeddedSetup` set to `true`, it runs in the same origin as the host application. JavaScript injected through this vulnerability then executes with full access to the host application's cookies, DOM, and same-origin APIs, so an attacker can reach all resources of the host application, not only those of Unblu. This expanded blast radius is why the critical rating applies to on-premises deployments that use this configuration.

In the fix versions, Unblu validates the `redirectOnFailure` value through the same redirect filter that already applies to other user-supplied redirect targets. After upgrading, review your redirect filter configuration. The defaults restrict redirects to the current origin (`document.location.origin`):

```properties
com.unblu.redirectFilter.order=WHITE_BLACK
com.unblu.redirectFilter.black=[]
com.unblu.redirectFilter.white=[]
```

Workarounds

Unblu Cloud customers don't need to take any action. In Unblu Cloud, the Content Security Policy (CSP) is enabled by default, and web application firewall (WAF) rules block all attempts to exploit this vulnerability.

If you can't upgrade an on-premises deployment immediately, apply at least one of the following measures to prevent exploitation:

- Enable the built-in CSP by setting `com.unblu.contentsecuritypolicy.mode` to `on`. The CSP blocks the injected JavaScript from executing. This protection is only effective on Unblu Spark 7.56.2 and later in the 7.x series, and 8.19.1 and later in the 8.x series, where Unblu removed `unsafe-inline` from the script source policy. Earlier versions ship a CSP that still permits `unsafe-inline` and aren't protected by this workaround.
- Configure a WAF rule to block requests whose `redirectOnFailure` parameter carries a malicious payload, in particular values that contain a `javascript:` scheme or obfuscated variants of it.

UBL-2025-002: File upload functionality possible even when disabled

Published on: 2025-04-22

CVE ID: CVE-2025-3518

Severity: 5.3 Medium

Credits: Andrei Dabrakou, Citadelo

Summary

User can upload files to a conversation even if the file upload functionality is disabled.

Affected components and versions

- Unblu Spark (v8): 8.12.1 and earlier
- Unblu Spark (v7): 7.53.4 and earlier

Fix versions

- Unblu Spark (v8): 8.13.1

- Unblu Spark (v7): 7.54.1

Details

The file upload functionality can be configured in such a way that it's only enabled or disabled for specific use cases.

If the functionality is disabled in at least one use case, Unblu nevertheless accepts and completes API requests to upload files.

When the user uploads a file, it's processed like any other uploaded file: it must still pass any file interceptors in place, and only allowed file types are accepted.

UBL-2025-001: Replace uploaded files knowing the file upload ID

Published on: 2025-04-22

CVE ID: CVE-2025-3519

Severity: 7.0 High

Credits: Andrei Dabrakou, Citadelo

Summary

Participants of a conversation can replace a file in the conversation without changing the file name, provided they know the file upload ID.

Affected components and versions

- Unblu Spark (v8): 8.12.1 and earlier

Fix versions

- Unblu Spark (v8): 8.13.1

Details

Every uploaded file in Unblu is assigned a randomly generated Universally Unique ID (UUID).

If a participant in a conversation gets access to a file's UUID, they can use this information to replace the file without changing the file's name, the file's details, or the name of the user who uploaded the file.

The participant needn't be a participant in the conversation the file was uploaded to. They can be a participant in any conversation.

When the participant replaces the uploaded file, the new file is processed like any other uploaded file: it must still pass any file interceptors in place, and only allowed file types are accepted.

Online Cookies Notice Cookie Settings Imprint © 2026 Unblu Inc.
All Rights Reserved
swiss made software