INTELLIGENCE BRIEFING
Security Command Center
TLP:CLEAR
2026-03-29 18:33 UTC

# No-Code Platform Bubble.io Abused as Microsoft 365 Phishing Infrastructure via AI-Generated Shadow DOM Obfuscation

**THREAT CAMPAIGN** | **HIGH** | CVSS 7.5

| | |
|---|---|
| **SCC Item ID** | SCC-CAM-2026-0105 |
| **Type** | Threat Campaign |
| **Severity** | HIGH |
| **CVSS Base Score** | 7.5 |
| **Affected Products** | Microsoft 365 (credential theft target); Bubble.io (abused hosting platform); Cloudflare (used as cover layer by threat actors) |
| **Published** | 2026-03-25 |
| **Discovery Source** | Rss |

## Executive Summary

Threat actors are exploiting Bubble.io, a legitimate no-code application platform, to host Microsoft 365 credential-harvesting pages that bypass standard email security controls. Because Bubble.io holds trusted domain reputation, phishing links evade URL reputation filters and email gateway allow-lists that would block dedicated attacker infrastructure. Kaspersky researchers (as reported by BleepingComputer) warn the technique is likely to be integrated into Phishing-as-a-Service kits, which would significantly expand the volume and reach of these attacks against any organization using Microsoft 365.

## Technical Analysis

Attackers host phishing redirectors on Bubble.io (*.bubble.io subdomains), using the platform's trusted domain status to pass email security reputation checks. AI-generated Shadow DOM structures obfuscate page content, defeating static HTML analysis and signature-based parsing by email security tools, mapped to CWE-116 (Improper Encoding/Escaping) and CWE-79 (Improper Neutralization of Input, XSS context). Cloudflare services layer over the origin infrastructure to obscure hosting provenance (CWE-601, Open Redirect). The attack chain aligns with MITRE ATT&CK T1566.002 (Spearphishing Link), T1027/T1027.010 (Obfuscated Files or Information / Command Obfuscation), T1583.006 (Acquire Infrastructure: Web Services), T1598.003 (Spearphishing Link for credential access), T1539 (Steal Web Session Cookie), and T1557 (Adversary-in-the-Middle). No CVE is assigned; this is a platform-abuse campaign, not a software vulnerability. The technique mirrors a documented pattern: Huntress previously identified Railway.com PaaS abused for

M365 token replay attacks. Kaspersky assessed this technique is likely to propagate into PhaaS toolkits. Threat actor attribution is unverified and low-confidence; treat as unconfirmed pending IOC publication. No patch applies; mitigations are detection- and policy-based.

## Action Checklist

1. Step 1, Immediate: Add *.bubble.io to email gateway review queues and URL filtering watchlists; do not blanket-block without impact assessment, as legitimate Bubble.io applications may exist in your environment. Review current allow-list entries for no-code and PaaS platforms (bubble.io, railway.app, glitch.me, and equivalents) and validate each entry is still warranted.
2. Step 2, Detection: Query email gateway and proxy logs for user clicks on bubble.io URLs over the past 90 days. Prioritize any that redirected to Microsoft login-styled pages or generated authentication token activity. Check Azure AD / Entra ID sign-in logs for anomalous authentications following bubble.io link clicks, look for new device, new geography, or impossible-travel indicators.
3. Step 3, Assessment: Identify all Microsoft 365 accounts that clicked bubble.io links in the review window. For any account with post-click authentication anomalies, treat as potentially compromised: review OAuth token grants, active sessions, and mail forwarding rules. Reference the Huntress Railway.com blog (cited in sources) for parallel token replay campaign indicators; apply equivalent pattern-matching logic to your logs to detect similar behavior originating from Bubble.io access events.
4. Step 4, Communication: Brief security operations and helpdesk teams on the phishing vector so they can correctly triage user-reported suspicious emails. If compromised accounts are identified, notify affected users and escalate per your incident response playbook. Inform the Microsoft 365 administration team to audit OAuth grants and conditional access policy coverage.
5. Step 5, Long-term: Review email security tool configuration for Shadow DOM parsing capability and static HTML analysis limitations; engage vendor for guidance on handling AI-generated DOM obfuscation. Establish a periodic review process for trusted-domain allow-list entries covering legitimate-but-abusable platforms. Update phishing awareness training to include scenarios where links appear to originate from legitimate application platforms rather than obvious attacker domains.

## IR / Forensic Enrichment

| | |
|---|---|
| **Triage Priority** | URGENT |
| **Escalation Criteria** | If more than 5 accounts show both bubble.io link clicks AND anomalous sign-ins (new device, impossible travel, or token grants), or if evidence of mailbox forwarding rule creation is discovered, escalate to Chief Information Security Officer and external incident response firm within 4 hours. |
| **Recovery Notes** | For each confirmed compromised account: force password reset, revoke all active OAuth tokens from Azure AD portal, disable inactive sessions, and review mailbox rules for forwarding redirects (delete any unauthorized rules). Re-enable the account only after conditional access policy confirms MFA enrollment and device compliance. For bulk recovery, use PowerShell: Get-AzureADUser -Filter "UserType eq 'Member'" \| Revoke-AzureADUserAllRefreshToken to invalidate all tokens organization-wide if scope of compromise is >10% of user base. After 48 hours, scan affected accounts for shadow mailbox rules and email forwarding via Get-InboxRule and Get-Mailbox -IncludeUnsecured, then monitor sign-in logs for 30 days for re-compromise indicators. |

| | |
|---|---|
| **Forensic Artifacts** | Microsoft 365 / Azure AD sign-in logs (CSV export with timestamp, user, device, location, risk level) \| Email gateway click/redirect logs and URL reputation checks (Proofpoint Forensics module, Mimecast audit trail, or Defender for Office 365 threat explorer) \| Browser history and cookies from user workstations (Chrome, Edge, Firefox profile directories; specifically cache of login.microsoft.com variants and bubble.io subdomains) \| PowerShell/Azure AD audit logs for OAuth token grants and mailbox rule changes (Get-AzureADAuditDirectoryLogs, Get-AdminAuditLogEvent, Get-InboxRule output with timestamps) \| Network proxy/firewall logs for outbound HTTPS connections to bubble.io domains, redirects to Microsoft login pages, and POST requests to credential harvesting endpoints (including DNS resolution logs and SSL/TLS certificates presented during handshake) |

### Per-Action IR Details

**Step 1 — Immediate: Add *.bubble.io to email gateway review queues and URL filtering watchlists; do not blanket-block without impact assessment, as legitimate Bubble.io applications may exist in your environment. Review current allow-list entries for no-code and PaaS platforms (bubble.io, railway.app, glitch.me, and equivalents) and validate each entry is still warranted.**

**NIST Phase:** Preparation

**Reference:** NIST 800-61r3 §2.1 (preparation phase: tools and processes)

**Controls:** NIST 800-53 CA-7 (continuous monitoring), NIST 800-53 SC-7 (boundary protection), CIS 6.1 (log collection), CIS 6.2 (centralized logging)

**Compensating:** For teams without advanced URL filtering: export allow-list entries to CSV, cross-reference against your business application inventory (asset management system or manual vendor list), then flag *.bubble.io, *.railway.app, *.glitch.me for manual email analyst review before delivery. Implement a rule in your email gateway to tag (not block) bubble.io URLs for 14 days while you audit usage: most platforms support header-based tagging or conditional rules without requiring third-party plugins.

**Evidence:** Before implementing allow-list changes, export current proxy/gateway logs for the past 90 days to preserve baseline traffic: export from your email gateway's URL logging table (e.g., ProofPoint, Mimecast, Microsoft Defender for Office 365), capture firewall access logs for *.bubble.io, and document the date/time of the allow-list review and any entries modified. Preserve original allow-list configuration in version control or a timestamped backup file.

**Step 2 — Detection: Query email gateway and proxy logs for user clicks on bubble.io URLs over the past 90 days. Prioritize any that redirected to Microsoft login-styled pages or generated authentication token activity. Check Azure AD / Entra ID sign-in logs for anomalous authentications following bubble.io link clicks — look for new device, new geography, or impossible-travel indicators.**

**NIST Phase:** Detection Analysis

**Reference:** NIST 800-61r3 §3.1 (detection: identifying anomalies in system logs)

**Controls:** NIST 800-53 SI-4 (information system monitoring), NIST 800-53 AU-6 (audit review, analysis, and reporting), CIS 4.1 (event logging), CIS 8.2 (user account monitoring)

**Compensating:** Without a SIEM: query email gateway logs directly via CLI or web UI with a 90-day filter for domain:bubble.io, export to CSV, and manually filter for rows where 'action' = CLICK or REDIRECT. For Azure AD sign-ins without advanced analytics, download the raw sign-in log CSV from Azure Portal > Azure AD > Sign-in logs, then use Excel/Python to identify rows where SigninTime - ClickTime < 5 minutes AND (DeviceInfo differs from baseline OR UserLocation differs from last 10 logins). Document the query timestamp and export date for evidence preservation.

**Evidence:** Preserve email gateway click/redirect logs with full URL and timestamp (e.g., Proofpoint Forensics Export or Mimecast Log API dump). Capture Azure AD sign-in logs as CSV from https://portal.azure.com/#blade/Microsoft_AAD_IAM/UsersManagementMenuBlade/SignInLogs (filter by date range 90 days). Export browser history from affected users' workstations (C:\Users\[USER]\AppData\Local\Google\Chrome\User Data\Default\History for Chrome; similar paths for Edge/Firefox). Preserve HTTP proxy access logs showing

GET/POST requests to bubble.io domains with response codes and user-agent strings.

**Step 3 — Assessment: Identify all Microsoft 365 accounts that clicked bubble.io links in the review window. For any account with post-click authentication anomalies, treat as potentially compromised: review OAuth token grants, active sessions, and mail forwarding rules. Cross-reference against Huntress's Railway.com token replay indicators to assess whether parallel campaigns targeted the same accounts.**

**NIST Phase:** Detection Analysis

**Reference:** NIST 800-61r3 §3.2 (analysis phase: determining scope and impact of compromise)

**Controls:** NIST 800-53 AC-2 (account management), NIST 800-53 SI-4 (information system monitoring), NIST 800-53 IA-4 (identifier management), CIS 6.1 (centralized logging), CIS 5.3 (access control lists on shared resources)

**Compensating:** Without Huntress integration: manually audit each identified account by querying the Microsoft 365 admin portal. For each account, check (1) Azure AD > Users > [Account] > Devices to verify all registered devices are expected; (2) Microsoft 365 admin center > Settings > Mail forwarding rules by running PowerShell: Get-Mailbox -ResultSize Unlimited | Get-InboxRule | Select Name, Description, ForwardTo; (3) OAuth grants via Azure AD > App registrations > Consent and permissions > [filter by risk]. Cross-reference the identified accounts against public IoC feeds (e.g., Shodan, GreyNoise) and check for leaked credentials via Have I Been Pwned API.

**Evidence:** Export all identified compromised accounts to a master evidence spreadsheet. For each account, capture: (1) OAuth token grant list from Azure AD portal (screenshot + CSV export of Permissions granted date, app name, scope); (2) Active sessions from Microsoft 365 > Users > Active users > [Account] > Sessions (screenshot of last 30 days); (3) Mailbox forwarding rules output from Get-InboxRule PowerShell command (run as-is, timestamp the output); (4) Anomalous sign-in details from Azure AD > Sign-in logs for that user (filter to 24 hours post-click, export CSV). Preserve browser cookies from affected workstations (e.g., /var/lib/firefox/[PROFILE]/cookies.sqlite or Chrome's Local State file) to identify compromised OAuth tokens.

**Step 4 — Communication: Brief security operations and helpdesk teams on the phishing vector so they can correctly triage user-reported suspicious emails. If compromised accounts are identified, notify affected users and escalate per your incident response playbook. Inform the Microsoft 365 administration team to audit OAuth grants and conditional access policy coverage.**

**NIST Phase:** Containment

**Reference:** NIST 800-61r3 §3.3 (containment: preventing further compromise)

**Controls:** NIST 800-53 IR-6 (incident reporting), NIST 800-53 SI-4 (information system monitoring), NIST 800-53 AC-2 (account management), CIS 17.1 (incident response training and testing)

**Compensating:** Without a formal incident response platform: draft a one-page alert email template describing the bubble.io phishing campaign, expected indicators (bubble.io URLs, Microsoft login-styled redirects), and triage steps for users who clicked (password reset, 2FA re-enrollment, check forwarding rules). Send to helpdesk and SOC; request confirmation of receipt within 2 hours. For compromised accounts, use a standardized notification email per your IR playbook (template: account name, incident reference ID, action required by user, helpdesk contact). For M365 admins, provide a bulleted list of conditional access policies to validate (e.g., 'Require MFA for high-risk sign-ins', 'Block legacy authentication', 'Block mob devices without Intune enrollment') and ask for confirmation of audit completion within 48 hours.

**Evidence:** Document all communications in a central incident log (e.g., timeline entry in Jira, ServiceNow, or plain text file). For each notified party, record: timestamp of notification, delivery method (email, Teams, phone), recipient name/role, and confirmation of receipt. Preserve all alert emails and briefing materials (template used, version number, date sent). For user notifications, maintain a list of account names notified, notification date/time, and user response (e.g., password reset timestamp from Azure AD audit logs).

**Step 5 — Long-term: Review email security tool configuration for Shadow DOM parsing capability and static HTML analysis limitations; engage vendor for guidance on handling AI-generated DOM obfuscation. Establish a periodic review process for trusted-domain allow-list entries covering legitimate-but-abusable platforms. Update phishing awareness training to include scenarios where links appear to originate from legitimate application platforms rather than obvious attacker domains.**

**NIST Phase:** Post Incident

**Reference:** NIST 800-61r3 §3.4 (post-incident activities: lessons learned and process improvement)

**Controls:** NIST 800-53 SI-4 (information system monitoring), NIST 800-53 AT-3 (role-based security training), NIST 800-53 PM-14 (testing, training, and monitoring), CIS 17.1 (incident response training and testing), CIS 14.6 (user security awareness training)

**Compensating:** Engage your email gateway vendor (Proofpoint, Mimecast, Defender for Office 365, etc.) via official support channels to request: (1) confirmation of Shadow DOM/AI obfuscation detection capabilities in their current platform version, (2) any available detection rules or ML models specifically trained on phishing-as-a-service campaigns. If vendor lacks native capability, implement a compensating control: deploy a web proxy rule (e.g., in Squid, pfSense, or cloud-native proxies) that parses HTML content for telltale phishing patterns (form tags targeting 'login.microsoft.com', script tags containing base64-encoded credential harvesters, SVG/Canvas elements used for obfuscation). For allow-list reviews, establish a quarterly calendar reminder to audit allow-list entries: export, cross-reference against your business application inventory, flag any entries without documented business justification, and track in a spreadsheet. For training, create a 5-minute phishing simulation scenario featuring a Bubble.io login page styled to mimic Microsoft 365 (use your email platform's built-in simulation tool or services like Knowbe4, PhishLabs); require completion within 30 days of campaign discovery and track results.

**Evidence:** Document vendor responses (support tickets, emails, dated correspondence) regarding Shadow DOM and AI obfuscation detection. Preserve before-and-after email gateway configurations (export configuration files or take timestamped screenshots of rules/policies). Maintain a spreadsheet of allow-list review decisions with columns: [Domain, Business Owner, Justification, Review Date, Approved/Flagged]. Track phishing simulation completion rates and failure-to-click metrics in your email security platform's reporting dashboard (export report as PDF for baseline comparison). Archive training materials (simulation email template, results summary) in your incident documentation repository.

## Detection Guidance

Primary detection surfaces: email gateway logs, proxy/DNS logs, and Azure AD / Entra ID sign-in logs.

1. Email gateway: Search for inbound messages containing URLs matching the pattern *bubble.io/* with redirect parameters (look for query strings containing 'url=', 'redirect=', 'next=', or base64-encoded strings in the path). Flag messages where the bubble.io URL is the sole or primary call-to-action link.

2. Proxy and DNS logs: Query for DNS resolution or HTTP requests to *.bubble.io originating from mail client processes (Outlook, browser following email link). Correlate timestamps: a bubble.io request followed within 60 seconds by a request to login.microsoftonline.com or a lookalike domain is a high-priority indicator.

3. Azure AD / Entra ID: Use the Sign-in Logs workbook or KQL query, filter for 'RiskLevelDuringSignIn = high' OR 'RiskLevelAggregated = high' with a preceding referrer from bubble.io. Also query for new OAuth application grants within 24 hours of a bubble.io access event.

4. Behavioral indicator: Users reporting Microsoft 365 login prompts they did not expect after clicking a link, especially if the login page URL is not login.microsoftonline.com.

5. Shadow DOM obfuscation note: Static HTML inspection of the phishing page may return minimal or misleading content. If an analyst is reviewing a suspected URL, dynamic rendering (browser-based inspection with JavaScript execution) is required; static source review will not reveal the credential-harvesting form. Treat inconclusive static analysis as a reason to escalate, not clear.

As of the Kaspersky research publication, no confirmed IOCs (domains, IPs, hashes) have been published. Monitor Kaspersky Secure List and the cited BleepingComputer and Huntress sources for IOC publication; update detection rules upon release.

## Indicators of Compromise

| Type | Value | Context | Confidence |
|---|---|---|---|
| DOMAIN | `*.bubble.io` | Bubble.io platform subdomains used as phishing redirector hosting. Pattern indicator only — not all bubble.io domains are malicious. Use for log review and anomaly correlation, not blanket blocking. | LOW |

## Framework Mappings

### MITRE-ATTACK

- **T1078.004** — Cloud Accounts
- **T1059.007** — JavaScript
- **T1539** — Steal Web Session Cookie
- **T1557** — Adversary-in-the-Middle
- **T1598.003** — Spearphishing Link
- **T1598** — Phishing for Information
- **T1027** — Obfuscated Files or Information
- **T1189** — Drive-by Compromise
- **T1566.002** — Spearphishing Link
- **T1583.006** — Web Services
- **T1027.010** — Command Obfuscation

### NIST-800-53R5

- **SI-3** — Malicious Code Protection
- **SI-4** — System Monitoring
- **AT-2** — Literacy Training and Awareness
- **SC-7** — Boundary Protection
- **SI-8** — Spam Protection
- **SI-10** — Information Input Validation

### OWASP-TOP10-2021

- **A03:2021** — Injection

### CIS-V8

- **16.10**
- **6.3** — Require MFA for Externally-Exposed Applications
- **14.2** — Train Workforce Members to Recognize Social Engineering Attacks

### ISO-27001-2022

- **A.8.28** — Secure coding
- **A.5.34** — Privacy and protection of personal information
- **A.5.23** — Information security for use of cloud services

**HIPAA-SECURITY**

- **164.312(d)** — Person or Entity Authentication
- **164.308(a)(5)(i)** — Security Awareness and Training

**SOC2-TSC**

- **CC6.1** — Logical access security software, infrastructure, and architectures

## MITRE ATT&CK Mapping

| Technique ID | Technique Name | Tactic |
|---|---|---|
| T1078.004 | Cloud Accounts | Defense-Evasion |
| T1059.007 | JavaScript | Execution |
| T1539 | Steal Web Session Cookie | Credential-Access |
| T1557 | Adversary-in-the-Middle | Credential-Access |
| T1598.003 | Spearphishing Link | Reconnaissance |
| T1598 | Phishing for Information | Reconnaissance |
| T1027 | Obfuscated Files or Information | Defense-Evasion |
| T1189 | Drive-by Compromise | Initial-Access |
| T1566.002 | Spearphishing Link | Initial-Access |
| T1583.006 | Web Services | Resource-Development |
| T1027.010 | Command Obfuscation | Defense-Evasion |

## Sources

| Source | URL | Tier |
|---|---|---|
| **Security News** | https://www.bleepingcomputer.com/news/security/bubble-ai-app-builde... | T3 |
| | https://www.bleepingcomputer.com/news/security/bubble-ai-app-builde... | T3 |
| | https://www.bleepingcomputer.com/news/security/anthropic-claims-of-... | T3 |
| | https://www.bleepingcomputer.com/news/security/google-gemini-flaw-h... | T3 |

| Source | URL | Tier |
|---|---|---|
| **Threat Actors Abuse Railway.com PaaS as Microsoft 365 Token ...** | https://www.huntress.com/blog/railway-paas-m365-token-replay-campaign | **T3** |



Generated 2026-03-29 18:33 UTC by TJS Security Command Center