Voatz

Comments & Remediations

1. TOB-VOATZ-014 (Device IDs not validated against inner request device IDs)
For endpoints that are sensitive, Voatz believes that the additional session level checks in place in the code provide sufficient protection against such a threat.
For endpoints that are not sensitive, Voatz accepts the risk presented here and believes its layered security protocols will detect misuse.

2. TOB-VOATZ-017 (Amazon admin password)
This unused test code has been removed from the repository.

4. TOB-VOATZ-013 (Secrets stored as environment variables)
Voatz accepts the risk presented here and believes that its internal controls provide sufficient safeguards to prevent misuse.

5. TOB-VOATZ-010 (API for onboarding workflow)
Given the nature of the current election pilots, Voatz is comfortable with the current approach and plans to address scalability challenges as part of the version 2 of its platform.

6. TOB-VOATZ-020 (Receipt and affidavit filename collisions)
This is mitigated by ensuring single-use audit tokens that cannot be reused again.

7. TOB-VOATZ-022 (A voter can unregister another voter's device)
This has been addressed by permanently enforcing the relevant parameter value.

9. TOB-VOATZ-028 (Voatz backend SSL key has a subdomain wildcard)
Voatz accepts the risk presented here and plans to re-evaluate this at a later time.

10. TOB-VOATZ-046 (Clients can specify their own audit token)
Voatz accepts the risk presented here and believes the other controls in place (such as single use audit tokens) in the system provide sufficient safeguards to prevent misuse.

11. TOB-VOATZ-047 (Test parameters in the registration APIs)
This test code has been removed from the repository.

13. TOB-VOATZ-018 (Session timeout validation)
A memcached session entry is ejected when the TTL or TimeToLive expires. This setting is configurable server side and is used to control the duration of a valid session. This renders the MaxIdleTime setting superfluous and it is for this reason the code reading the MaxIdleTime setting was commented out.

14. TOB-VOATZ-015 (Receipt encryption is weak)
Voatz is working on addressing this as part of its upcoming release.

15. TOB-VOATZ-019 (Insufficient device ID validation)
Voatz is working on adding good-form validation as part of its upcoming release.

16. TOB-VOATZ-035 (Potential resource exhaustion)
Voatz accepts the risk presented here and believes its layered security protocols will detect this early and stop the misuse.

17. TOB-VOATZ-030 (Resource exhaustion)
Voatz accepts the risk presented here and believes its layered security protocols will detect this early and stop the misuse.

18. TOB-VOATZ-029 (Zimperium checks)
Voatz believes that its 3-way off channel check will detect attempts to bypass Zimperium as such a check is not visible to an attacker.

19. TOB-VOATZ-024 (AES-GCM key/nonce/tag breaks authenticity)
This has been addressed as part of the fix for TOB-VOATZ-011.

20. TOB-VOATZ-004 (Unauthenticated ECDH)
This is being addressed as part of the noise protocol implementation in an upcoming release.

22. TOB-VOATZ-033 (OSCP Stapling)
This has been enabled on the relevant Voatz servers.

23. TOB-VOATZ-027 (Empty ballots are not recorded)
Voatz has addressed this in the audit documentation.

24. TOB-VOATZ-016 (Database root credentials)
These were old local test credentials from a few years ago and are no longer used.

25. TOB-VOATZ-021 (Signed voter affidavits are sent to an administrative email)
This finding is not relevant. Firstly, this is required per the legal guidelines of the election jurisdictions. See sample affidavit for reference. Secondly, the destination email is provided by the jurisdiction. The email address in your snippet is just a placeholder. Thirdly, our pilot jurisdictions already allow eligible absentee voters to return ballots via email or efax (~remember Voatz is an additional method that is being piloted) and have their own practices, procedures in terms of handling spam, etc. Lastly, Voatz servers send these emails from a whitelisted email address using a dedicated IP address and using a service that is protected using DMARC, DKIM, SPF.

26. TOB-VOATZ-012 (AES-GCM AAD usage is non-standard)
Voatz is comfortable with its AAD usage and believes that it aids its security protocols.

27. TOB-VOATZ-005 (Session cookie offset)
Voatz accepts the risk here and plans to re-evaluate this in a future release.

28. TOB-VOATZ-048 (Encrypted application data is brute forceable)
Voatz is enhancing this functionality as part of its upcoming release. The salt is actually stored in an encrypted shared preferences file so there is partial mitigation in place already.

29. TOB-VOATZ-025 (PDKF2 provides insufficient security)
This has been partially addressed. The remainder is being addressed as part of the updates for TOB-VOATZ-048.

31. TOB-VOATZ-003 (Android release build signing keys)
Voatz accepts the risk presented here and believes the frequent key rotation, other controls in place provide sufficient safeguards in the short term.

33. TOB-VOATZ-008 (Insufficient Android device ID construction)
This was addressed as part of the earlier fix. The reregistration upon device reset is a mandatory part of the user workflow and users are advised about the same via help messages, other tutorials.

34. TOB-VOATZ-037 (Android client doesn't use SafetyNet Attestation API)
This is being added as part of the upcoming release.

35. TOB-VOATZ-045 (Android client doesn't use SafetyNet Verify API)
This is being added as part of the upcoming release.

36. TOB-VOATZ-026 (Certificate pinning is only configured for main domain)
Voatz accepts the risk here and is addressing this as part of an upcoming release.

37. TOB-VOATZ-034 (No explicit verification of Android security provider)
This is addressed via the Zimperium integration.

38. TOB-VOATZ-001 (Jumio Netverify API credentials stored in git)
These were a few years old and have since been rotated and are no longer stored in git.

39. TOB-VOATZ-002 (Google Services API key stored in git)
Voatz accepts the risk presented here and believes the frequent key rotation, other controls in place provide sufficient safeguards in the short term.

41. TOB-VOATZ-040 (iOS client doesn't disable custom keyboards)
Voatz accepts the risk here and plans to address this in an upcoming release.

42. TOB-VOATZ-042 (iOS client doesn't use system managed login input fields)
Voatz accepts the risk here and plans to re-evaluate this for an upcoming release.

43. TOB-VOATZ-043 (iOS keychain items)
Voatz has addressed this in its upcoming release.

44. TOB-VOATZ-041 (Cryptographic credential generation)
Due to an ongoing need to support certain iOS devices that cannot be upgraded beyond iOS 12.4.4, this change is currently pending and will be incorporated as soon as feasible.

45. TOB-VOATZ-044 (iOS client disables ATS)
Voatz accepts the risk presented here, as this is required to support the Zimperium integration.

46. TOB-VOATZ-038 (iOS client is vulnerable substitution attacks)
Voatz accepts the risk presented here and is planning to address this in a future release.

47. TOB-VOATZ-007 (iOS user can lose registration)
This is intentional and designed for security. It is similar to TOB-VOATZ-008 wherein the Android reregistration is enforced.

48. TOB-VOATZ-039 (iOS client is susceptible to URI scheme hijacking)
Voatz accepts the risk presented here and confirms that it is not using the URI scheme for any purposes.

Last Updated: March 10, 2020