Google Authenticator

Google Authenticatorเป็นโทเค็นซอฟแวร์ที่ใช้ในการตรวจสอบแบบสองขั้นตอนการให้บริการโดยใช้ครั้งเดียวรหัสผ่านตามเวลาอัลกอริทึม (TOTP; ระบุไว้ในRFC 6238) และHMAC ตามขั้นตอนวิธีการรหัสผ่านครั้งเดียว (HOTP; ระบุไว้ในRFC 4226) สำหรับผู้ใช้ตรวจสอบการใช้งานโทรศัพท์มือถือโดยGoogle [2]

Google Authenticator

(s)	Google
การเปิดตัวครั้งแรก	20 กันยายน 2010 [1]
กรุ	github .com / google / google-authenticator
ระบบปฏิบัติการ	Android , iOS , BlackBerry OS
เวที	โทรศัพท์มือถือ
การอนุญาต	กรรมสิทธิ์ (เวอร์ชั่นก่อนหน้านี้อยู่ภายใต้Apache License 2.0)
เว็บไซต์	github .com / google / google-authenticator-libpam

libpam

เมื่อเข้าสู่เว็บไซต์ที่รองรับ Authenticator (รวมถึงบริการของ Google) หรือใช้แอปพลิเคชันของบุคคลที่สามที่รองรับ Authenticator เช่นผู้จัดการรหัสผ่านหรือบริการโฮสติ้งไฟล์ Authenticator จะสร้างรหัสผ่านครั้งเดียวหกถึงแปดหลักที่ผู้ใช้ต้องป้อนเพิ่มเติม รายละเอียดการเข้าสู่ระบบตามปกติ

รุ่นก่อนหน้านี้ของซอฟต์แวร์ที่เป็นโอเพนซอร์สแต่รุ่นต่อมาเป็นกรรมสิทธิ์ [3]

กรณีการใช้งานทั่วไป

To use Authenticator, the app is first installed on a smartphone. It must be set up for each site with which it is to be used: the site provides a shared secret key to the user over a secure channel, to be stored in the Authenticator app. This secret key will be used for all future logins to the site.

To log into a site or service that uses two-factor authentication and supports Authenticator, the user provides username and password to the site, which computes (but does not display) the required six-digit one-time password and asks the user to enter it. The user runs the Authenticator app, which independently computes and displays the same password, which the user types in, authenticating their identity.

With this kind of two-factor authentication, mere knowledge of username and password is not sufficient to break into a user's account; the attacker also needs knowledge of the shared secret key, or physical access to the device running the Authenticator app. An alternative route of attack is a man-in-the-middle attack: if the computer used for the login process is compromised by a trojan, then username, password and one-time password can be captured by the trojan, which can then initiate its own login session to the site or monitor and modify the communication between user and site.

คำอธิบายทางเทคนิค

The service provider generates an 80-bit secret key for each user (whereas RFC 4226 §4 requires 128 bits and recommends 160 bits).[4] This is provided as a 16, 26 or 32 character base32 string or as a QR code. The client creates an HMAC-SHA1 using this secret key. The message that is HMAC-ed can be:

- the number of 30-second periods having elapsed since the Unix epoch (TOTP); or
- the counter that is incremented with each new code (HOTP).

A portion of the HMAC is extracted and converted to a six-digit code.

Pseudocode for one-time password (OTP)

```
 function GoogleAuthenticatorCode(string secret)
     key := 5B5E7MMX344QRHYO
     message := floor(current Unix time / 30)
     hash := HMAC-SHA1(key, message)
     offset := last nibble
```

```
of hash
      truncatedHash := hash[offset..offset+3]  //4 bytes starting at the offset
      Set the first bit of truncatedHash to zero //remove the most significant bit
      code := truncatedHash mod 1000000
      pad code with 0 from the left until length of code is 6
      return code
```

ซอฟต์แวร์ตรวจสอบสิทธิ์อื่น ๆ

The Google Authenticator app for Android was originally open source, but later became proprietary.[3] Google made earlier source for their Authenticator app available on its GitHub repository; the associated development page states:

> *"This open source project allows you to download the code that powered version 2.21 of the application. Subsequent versions contain Google-specific workflows that are not part of the project."*[5]

Following Google Authenticator ceasing to be open source, a free-software clone named FreeOTP[6][3] was created, predominantly a fresh rewrite but including some code from the original.Google provides Android,[7] BlackBerry, and iOS[8] versions of Authenticator.

Several other versions of authentication software are available. Those that use TOTP and HMAC in addition to other two-factor authentication can authenticate with the same sites and processes as Google Authenticator. Some of the listed software is available in versions for several platforms.

- Windows Phone 7.5/8/8.1/10: Microsoft Authenticator,[9] Virtual TokenFactor[10]
- Windows Mobile: Google Authenticator for Windows Mobile[11]
- Java CLI: Authenticator.jar[12]
- Java GUI: JAuth,[13] FXAuth[14]
- J2ME: gauthj2me,[15] lwuitgauthj2me,[16] Mobile-OTP (Chinese only),[17] totp-me[18]
- Palm OS: gauthj2me[19]
- Python: onetimepass[20] pyotp[21]
- PHP: GoogleAuthenticator.php[22]
- Ruby: rotp,[23] twofu[24]
- Rails: active_model_otp[25]

- webOS: GAuth[26]
- Windows: gauth4win,[27] MOS Authenticator,[28] WinAuth[29]
- .NET: TwoStepsAuthenticator[30]
- HTML5: html5-google-authenticator[31]
- MeeGo/Harmattan (Nokia N9): GAuth[32]
- Sailfish OS: SGAuth,[33] SailOTP[34]
- Apache: Google Authenticator Apache Module[35]
- PAM: Google Pluggable Authentication Module,[5] oauth-pam[36]
- Backend: LinOTP (Management Backend implemented in python)
- Chrome/Chrome OS: Authenticator[37]

- Multi-platform: Twilio Authy [38]
- Multi-platform: Duo Mobile [39]
- OTP Auth[40]
- privacyIDEA Authentication System.
- Multi-platform: LastPass Authenticator
- Android: andOTP

ดูเพิ่มเติม

- Multi-factor authentication
- HMAC-based One-time Password algorithm

อ้างอิง

1. *"Google Is Making Your Account Vastly More Secure With Two-Step Authentication - TechCrunch"* .

TechCrunch. 2010-09-20. Retrieved 2016-03-12.

2. *"GitHub - google/google-authenticator: Open source version of Google Authenticator (except the Android app)" . GitHub. Google. "These implementations support the HMAC-Based One-time Password (HOTP) algorithm specified in RFC 4226 and the Time-based One-time Password (TOTP) algorithm specified in RFC 6238 ."*

3. *Willis, Nathan (22 January 2014)."FreeOTP multi-factor authentication ". LWN.net. Retrieved 10 August 2015.*

4. *https://tools.ietf.org/html/rfc4226#section-4*

5. *"google-authenticator - Two-step verification - Google Project Hosting"* .

6. *"FreeOTP"* .

7. *https://play.google.com/store/apps/details?id=com.google.android.apps.authenticator2* *A*

8. *"Google Authenticator" . App Store.*

9. *"Authenticator" . 4 April 2013.*

10. *"Virtual TokenFactor" . 26 February 2012.*

11. *"[APP]Google Authenticator for Windows Mobile"* . *XDA Developers.*

12. *"http://blog dot jamesdotcuff dot net"* .

13. *"mclamp/JAuth"* . *GitHub.*

14. *"kamenitxan/FXAuth"* . *GitHub.*

15. *"gauthj2me - Google Authentification in Java Mobile, j2me - Google Project Hosting"* .

16. *"lwuitgauthj2me - Google Authenticator for J2ME phones - Google Project Hosting"* .

17. *"chunlinyao / mobile-otp — Bitbucket"* .

18. *"totp-me - TOTP for Java ME - Google authenticator"* .

19. *"gauth.prc - gauthj2me - Google Authenticator for Palm OS (converted from java) - Google Authentification in Java Mobile, j2me - Google Project Hosting"* .

20. *"tadeck/onetimepass"* . *GitHub.*

21. *"pyotp/pyotp"* . *GitHub.*

22. *"chregu/GoogleAuthenticator.php"* . *GitHub.*

23. *"rotp - RubyGems.org - your community gem host"* .

24. *"ukazap/twofu"* . *GitHub.*

25. *"heapsource/active_model_otp"* . *GitHub.*

26. *"GAuth"* .

27. *"gauth4win - Google Authenticator for windows - Google Project Hosting"* .

28. *"MOS Authenticator Home"* .

29. *"winauth - Windows Authenticator for Battle.net / World of Warcraft / Guild Wars 2 / Glyph / WildStar / Google / Bitcoin - Google Project Hosting"* .

30. *"glacasa/TwoStepsAuthenticator"* . *GitHub*.

31. *"gbraad/html5-google-authenticator"* . *GitHub*.

32. *Techtransit*. *"Nokia Store: Download GAuth and many other games, wallpaper, ringtones and mobile apps on your Nokia phone"* .

33. *"SGAuth"* .

34. *"SailOTP"* .

35. *"google-authenticator-apache-module - Apache Module for Two-Factor Authentication via Google Authenticator - Google Project Hosting"* .

36. *"oauth-pam - PAM for use with OAuth Websites - Google Project Hosting"* .

37. *"Authenticator"* .

38. *"Authy"* . *App Store.*

39. *"Duo Mobile"* . *App Store.*

40. *"OTP Auth"* . *App Store.*

ลิงค์ภายนอก

is article's use of external links may not follow Wikipedia's policies or guide'˙

Learn more

- Google Authenticator on Google Help

- Google Authenticator (Android) and Google Authenticator (other) legacy source code on GitHub
- Google Authenticator PAM module source code on GitHub
- Google Authenticator implementation in Python on Stack Overflow
- Authenticator on F-Droid
- Django-MFA Implementation Using Google Authenticator - Django-mfa is a simple package to add extra layer of security to your django web application. It gives web app a randomly changing password as an extra protection.