Second factor authentication in LemonLDAP::NG

Xavier GUIMARD

Clément OUDOT

Pass the SALT 2018
MFA / 2FA / OTP / U2F ?
Multi-factor authentication (MFA) is a method of confirming a user's claimed identity in which a user is granted access only after successfully presenting 2 or more pieces of evidence (or factors) to an authentication mechanism:

- knowledge (something they and only they know)
- possession (something they and only they have)
- inherence (something they and only they are)
Why you need it

- Passwords are still the main security token used to authenticate
- With GPU and rainbow tables it is more and more easy to crack a password
- A password base can be stolen
- Second factor authentication is hype (see Twitter, Github, LinkedIn...)
One-Time Password

- One-Time Password (OTP) is a password that is valid for only one login session or transaction
- Two standards:
  - HOTP (RFC 4226): HMAC-Based One-Time Password
  - TOTP (RFC 6238): Time-Based One-Time Password
- Rely on a secret shared between user and server
TOTP

- Shared secret key K
- T0: start time
- TI: time interval
- Time Counter TC = floor((unixtime(now) − unixtime(T0)) / TI)
- TOTP = Truncate( SHA1(K ⊕ 0x5c5c... ∥ SHA1(K ⊕ 0x3636... ∥ TC)) ) & 0x7FFFFFFF
- TOTP Value = TOTP mod 10d, where d is the desired number of digits of the one-time password
Using a TOTP

- Registration on client: shared key can be registered manually or using a QR code
- Server associates shared secret to user
- At next authentication, TOTP value is computed by client and server
Universal Second Factor

• Universal 2nd Factor (U2F) is an open authentication standard that strengthens and simplifies two-factor authentication using specialized USB or NFC devices.

• Managed by FIDO Alliance https://fidoalliance.org/

• Support in modern browsers:
  - Chrome/Chromium ≥ 38
  - Firefox:
    • 38 to 56 with U2F Support Add-on
    • 57 to 59, with “security.webauth.u2f” set to “true” in “about:config”
    • probably enabled by default for versions ≥ 60
  - Opera ≥ 40
Using U2F

- Registration: Token generates private/public keys and a handle and send public key and handle to server
- The server associates the public key and the handle to user
- At next authentication, server sends the handle and a crypto challenge and the U2F token signs the challenge and sends it back
U2F Registration

U2F Authenticator

generate:
key \( k_{pub} \)
key \( k_{priv} \)
handle \( h \)

\[ \rightarrow \]

\( a: \) challenge, origin, channel id, etc.

\( fc \)

\( k_{pub}, h, \) attestation cert, signature(\( a, fc, k_{pub}, h \))

\( s \)

\( fc, k_{pub}, h, \) attestation cert, \( s \)

store:
key \( k_{pub} \)
handle \( h \)

FIDO Client / Browser

check AppID

\( \rightarrow \)

AppID, challenge

\( \rightarrow \)

\( \rightarrow \)

Relying Party
U2F Authentication

U2F Authenticator

FIDO Client / Browser

Relying Party

retrieve:
key $k_{psv}$
from handle $h$;
cntr++

$h, a;\text{challenge, origin, channel id, etc.}$

$fc$

cntr, signature$(a, fc, cntr)$

$s$

handle, AppID, challenge

$h \quad a$

check AppID

retrieve
key $k_{pub}$
from handle $h$

check signature using
key $k_{pub}$

set cookie

cntr, fc, s
Implementation in LemonLDAP::NG
LemonLDAP::NG

- WebSSO, Access Control and Identity Provider
- Apache, Nginx, Node.js, Plack support
- Default protection by Handler, identity forwarded through HTTP headers
- Standard protocols: CAS, SAML and OpenID Connect
- Self services (password change, password lost, account registration)
- GPL License
- https://lemonldap-ng.org
2FA implementation

• New feature for 2.0 major version
• Possibility to add a second authentication step to any current authentication method
• A lot of possibilities to ask a second factor:
  – U2F tokens
  – TOTP (to use with FreeOTP, Google-Authenticator, …)
  – U2F-or-TOTP (enable both U2F and TOTP)
  – Yubikey tokens provide by Yubico
  – REST (Remote REST app)
  – External 2F (to call an external command)
Ask for second factor
User self registration

Your new TOTP key, please test it and enter the code

Name: MyTOTP
Code: 

Generate new key
Register

2ndFA Management
Go to portal
User self management

Choose your second factor

<table>
<thead>
<tr>
<th>Type</th>
<th>Name</th>
<th>Date</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>TOTP</td>
<td>MyTOTP</td>
<td>30/04/2018 à 11:17:39</td>
<td>Unregister</td>
</tr>
</tbody>
</table>

Go to portal
2FA sessions explorer