# BAB V
# Kesimpulan dan Saran

## 5.1 Kesimpulan

Berdasarkan pengujian yang dilakukan pada bab sebelumnya, Metode yang digunakan pada penelitian ini dapat melakukan autentikasi pada *client*. *Client* mendapatkan *credential* dari *authenticator* dan *client* berhasil melakukan autentikasi ke perangkat. *Credential* juga selalu berubah dan tidak dapat digunakan dua kali. *Authenticator* tidak menyimpan maupun mengirimkan *firmware*. Meskipun *attacker* mendapatkan *credential* dan dapat berkomunikasi dengan perangkat, *attacker* tetap tidak dapat melakukan autentikasi ke perangkat. *Credential* yang didapatkan oleh pengguna hanya dapat digunakan satu kali. Perangkat hanya dapat diperbaharui oleh satu *client* per proses. Cara ini menjamin tidak adanya *client* yang tidak memiliki *permission* atau wewenang dari server untuk melakukan FOTA *update* pada perangkat melakukan FOTA *update*.

Token yang dihasilkan perangkat dan diterima oleh *client* berhasil digunakan untuk memverifikasi pengguna. Token yang digunakan untuk memverifikasi *client* juga tidak dapat di buat ulang maupun dilakukan *brute-force*. Waktu dan *resource* yang dibutuhkan untuk melakukan hal tersebut terlalu besar sehingga dapat dikatakan tidak mungkin. Dikarenakan token tersebut juga membuat *signature* dari *firmware*, perangkat dapat mengetahu integritas dari *firmware* yang diterima. Hal ini mencegah perangkat dimasukin *malicious code*.

Kekurangan dari implementasi pada penelitian ini adalah keterbatasan kanal komunikasi yang digunakan. Waktu yang dibutuhkan untuk melakukan satu kali proses FOTA *update* cukup lama. Keterbatasan ini menunjukan dibutuhkannya protokol komunikasi khusus yang dapat mengakomodasi kebutuhan FOTA *update. Key* dari perangkat tidak dihasilkan dari fungsi didalam perangkat melainkan masih menggunakan fungsi dari luar.

## 5.2 Saran

Saran yang dapat disampaikan untuk penelitian ini dan penelitian selanjutnya adalah:

1. Penggunaan *microcontroller* yang dapat mengakses SRAM ataupun menggunakan eksternal SRAM untuk PUF
2. Menggunakan *microcontroller* yang memiliki ukuran *code* relatif lebih kecil
3. Membuat sebuah protokol yang dapat melakukan pertukaran *firmware* dan dapat di akomodasi oleh perangkat
4. Membuat *interface* untuk *client* yang dapat membantu melakukan FOTA *update* secara otomatis dan mudah.